

Article

Slice-Guided Components Detection and Spatial Semantics Acquisition of Indoor Point Clouds

Lijuan Wang ¹  and Yinghui Wang ^{2,*}

¹ School of Electronic Information Engineering, Xi'an Technological University, Xi'an 710021, China; wanglijuan@xatu.edu.cn

² School of Artificial Intelligence and Computer Science, Jiangnan University, Wuxi 214122, China

* Correspondence: wangyh@jiangnan.edu.cn

Abstract: Extracting indoor scene components (i.e., the meaningful parts of indoor objects) and obtaining their spatial relationships (e.g., adjacent, in the left of, etc.) is crucial for scene reconstruction and understanding. At present, the detection of indoor scene components with complex shapes is still challenging. To fix the problem, a simple yet powerful slice-guided algorithm is proposed. The key insight is that slices of indoor scene components always have similar profiles no matter if the components are simple-shaped or complex-shaped. Specifically, we sliced the indoor scene model into many layers and transformed each slice into a set of two-dimensional (2D) profiles by resampling. After that, we clustered 2D profiles from neighbor slices into different components on the base of spatial proximity and similarity. To acquire the spatial relationships between indoor scene components, an ontology was constructed to model the commonsense knowledge about the semantics of indoor scene components and their spatial relationships. Then the spatial semantics of the relationships between indoor scene components were inferred and a semantic graph of spatial relationship (SGSR) was yielded to represent them. The experimental results demonstrate that our method can effectively detect complex-shaped indoor scene components. The spatial relationships between indoor components can be exactly acquired as well.

Keywords: indoor scene components; point clouds; slices; spatial relationship; ontology



Citation: Wang, L.; Wang, Y. Slice-Guided Components Detection and Spatial Semantics Acquisition of Indoor Point Clouds. *Sensors* **2022**, *22*, 1121. <https://doi.org/10.3390/s22031121>

Academic Editor: Marcin Woźniak

Received: 14 December 2021

Accepted: 25 January 2022

Published: 1 February 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Components of indoor scenes are meaningful parts of indoor objects. Detecting the indoor scene components and acquiring their spatial relationships (e.g., adjacent, in the right of, in the left of, etc.) is one of the most important research problems in the computer vision and graphics community. As pointed out in many studies [1–3], the acquirement of indoor scene components and their spatial relationships will benefit many computer vision works such as indoor scene reconstruction and indoor scene understanding [4,5].

There are two main difficulties that arise during the detection of indoor scene components and their spatial relationships: (1) indoor scene components often have varied shapes and complex three-dimensional (3D) geometry. Moreover, the indoor scene components occlude each other. Thus, it is challenging to detect complex-shaped indoor scene components from point clouds; (2) due to the diverse internal structures of indoor objects and the messy arrangement of indoor objects, the spatial relationships between indoor scene components are complex, which makes it difficult to extract the spatial relationships between indoor scene components.

Most of the approaches [6–12] for the detection of indoor scene components concentrate on using primitive shapes (e.g., planes, cylinders, spheres, cuboids, etc.) to approximate the components and exploit 3D primitive shape segmentation algorithms such as Hough transforming [13,14] and Random Sample Consensus (RANSAC) [15,16] to detect indoor scene components. In these approaches, the primitive shape features of indoor

components are always pre-assumed, which is not suitable for complex-shaped indoor scene components.

Many methods [17–19] transform the scattered point clouds into 3D voxel grids and use spatial connectivity and geometric features to segment the indoor scene models. However, due to sparsity of the point clouds, the voxel grids may have empty voxels which leads to redundant computations. Moreover, it is difficult to select the appropriate resolution to accurately segment the components and preserve the boundaries due to the different scales of objects in the indoor scene model and the non-uniform point cloud density.

With the availability of large 3D datasets and the popularity of machine learning techniques, some data-driven segmentation methods [20–27] have been proposed for indoor scene components. In previous data-driven methods [20,21,27], indoor scene models are first segmented. Then the segmented results of the indoor scenes are classified into different components based on handcrafted features by machine-learning techniques, e.g., conditional random field (CRF), support vector machine (SVM) and so on. Motivated by directly learning features from input point clouds, the deep neural network has recently been exploited. Qi et al. [22] designed a novel type of neural network (PointNet) to provide a unified architecture for feature classification directly from point clouds. On the architecture, the labelling of components of objects is performed. Followed PointNet, other deep neural networks have been proposed, such as PointNet++ [28], the deep part induction network [23], the regularized graph convolutional neural network (RGCNN) [25], semantic part decomposition network [29] and so on. Although progress in detecting complex-shaped components is impressive, these methods are still inferior when it comes to discovering new components whose types are not covered in the training sets.

There are also other methods. Balado et al. [30] proposed a method to detect floor elements based on relative distances. In the references [31,32], surface patches of indoor scene models were merged into components according to the consistency of their local convexity or non-local geometric signature. Due to poor connectivity caused by missing parts and outliers of point clouds, convexity-based methods are not reliable for detecting the indoor scene components. The detection of complex-shaped indoor scene components is still challenging.

The extraction of the spatial relationships between indoor scene components lays a foundation for understanding the indoor scene in a way similar to the way that humans perceive the environment. Many methods [33–35] have been proposed to extract spatial relationships from scene images. In contrast with the spatial relationships in images, the spatial relationships in 3D point clouds are more complex [36,37], and the extraction of them is more challenging.

Recently, a few methods [38,39] have been proposed to extract spatial relationships from indoor point clouds based on machine learning techniques such as SVM and latent max-margin learning. However, it is difficult to build up a fixed parameter model for training due to the complexity of 3D spatial relationships. To fix the problem, Wald et al. [40] recently tried to use deep learning techniques to train and predict spatial relations. The deep learning-based method showed prospects in the extraction of certain spatial relationships. However, spatial relationships in 3D space are complex. It is difficult to obtain salient features between different spatial relationships and to effectively divide the spatial relationships into different categories based on the feature.

On the other hand, some approaches [41–44] have been proposed to extract spatial relationships from indoor point clouds based on prior spatial knowledge. For example, Zender et al. [42] presented an ontology to encode the spatial and functional knowledge of typical indoor environments. Suchan and Bhatt [43] adopted prior knowledge to extract commonsense spatio-temporal relations. Most existing knowledge-based methods have aimed to provide root navigation for indoor robots or model specific interactions between human and indoor objects. They mainly focused on the inter-object or human-centric spatial relationships. As a smaller-grained element of scenes, spatial relationships between the

indoor scene components are also affected by the structure of indoor objects. Accordingly, it is more difficult to detect the spatial relationships between indoor components.

In this paper, we present a framework to segment out the indoor scene components and detect their spatial relationships. Our method is based on a slice strategy. We are inspired by the methods in [45–47], where components of complex-shaped indoor objects were segmented based on the similarity of 2D profiles. Furthermore, our kernel insight lies in two points: (1) Slices of indoor scene components always have spatial proximity and similar profiles no matter if the components are simply or complex shaped. (2) The spatial topological relationships between indoor scene components can be effectively preserved by slicing the indoor scene layer by layer.

We use the slice strategy to obtain many slices of indoor scene models and convert each slice into a set of profiles, then merge the profiles of neighbor slices progressively into different components based on spatial proximity and similarity. Next, we geometrically establish relationships between the detected indoor scene components on the base of two geometric distances. Meanwhile, an ontology is built up to model the semantic knowledge about the spatial relationships between indoor scene components. The geometrically correlated indoor scene components are loaded to populate the ontology. Finally, the spatial semantics of the relationships are thereby inferred, and a semantic graph of spatial relationship (SGSR) is yielded to organize the indoor scene components and their spatial relationships.

The contributions of this paper can be summarized as follows:

- (1) We propose a slice-guided algorithm to detect complex-shaped indoor scene components from point clouds. The detected components are faithful to the meaningful parts of indoor objects;
- (2) We present a framework for modelling indoor scene components and their spatial relationship structure, which lays a foundation for the detection of following objects, semantic analysis, and understanding of indoor scenes.

The remainder of the paper is organized as follows. Section 2 presents a brief review of the extraction of indoor scene components and their spatial relationships. Section 3 gives the overview of the proposed method. Section 4 describes how to detect indoor scene components on the base of clustering of profiles. Section 5 elaborates on the inferring of spatial relationships between indoor scene components. The experimental results are presented in Section 6. The limitations of our method and proposals for future research are indicated in the last section.

2. Related Work

2.1. Detection of Indoor Scene Components

The extraction of indoor scene components from point clouds has received a lot of research interest in numerous works. Here we review the works of indoor scene components detection methods as follows. The methods can largely be classified into four types, i.e., the primitive shape proximity-based methods, the voxel grid-based methods, the data-driven methods, and other methods.

The primitive shape proximity-based methods approximate the indoor scene components with primitive shapes and use the primitive shape segmentation algorithms to detect indoor components. Rchnabel et al. [7] represented 3D semantic entities with configurations of basic shapes. Wang et al. [6] abstracted the sub-scenes with geometric primitives and their topological relationships with structural attributes. Li et al. [10] operated simultaneously on both the local and global aspects by fitting primitives locally while optimizing global relations iteratively. Hashemifar et al. [12] adapted a cuboid fitting algorithm for the mapping of indoor scenes. A limitation of the primitive shape proximity-based methods is that the commonly used primitive shape segmentation algorithms such as HOUGH transforming [13,14] and Random Sample Consensus (RANSAC) methods [15,16] are all based on statistical techniques. The segmentation results of these algorithms are randomly generated and the topological relationships between indoor scene components will be

lost. It hinders the following spatial relationship analysis between the components. Most importantly, the shapes of indoor scene components are complex and varied. When using these methods to detect the components, segments of indoor scenes (i.e., components) with complex shapes will not be detected effectively.

The voxel grid-based methods always adopt the 3D voxel grids representation of point clouds and perform segmentation on the simplified data structure. In [17], a point cloud was first voxelized by the octree. Then a K-means clustering algorithm was employed to realize super-voxel segmentation. Xu et al. [18] reported a novel strategy for segmenting 3D point clouds using a voxel structure and graph-based clustering with perceptual grouping laws. Lin et al. [19] proposed a new practice in super-voxel generation that adopted an adaptive resolution to preserve boundaries. It is challenging to accurately segment out components and preserve the boundaries in this kind of method.

The data-driven approaches often detect indoor scene components by training and applying a classifier to label the segments of indoor point clouds. For instance, Hausman et al. [21] pre-segmented a raw point cloud of a given scene using a part graph-based hashing algorithm, then an SVM-based classifier was trained by GRSD (Global Radius-based Surface Descriptor) feature and applied for the segments of point clouds. Recently, a few methods have employed deep learning to operate on indoor point clouds to segment out indoor scene components. Wang et al. [48] partitioned each object into smaller super-faces and each such super-face was associated with a vector of shape descriptors. Then must-link or cannot-link constraints between super-faces were added between super-faces through an active learning method. Qi et al. [22] designed a novel type of neural network (PointNet) that consisted of a Classification Network and Segmentation Network. PointNet well respects the permutation invariance of points in the input and can be directly used for indoor scene components segmentation from point clouds. Li et al. [23] introduced a new deep learning-based method to parse 3D objects into moving parts based on input static shape snapshots. Te et al. [25] used a regularized graph convolutional neural network (RGCNN) for the semantic segmentation of object parts. However, the data-driven methods need to label the amounts of the point clouds scanned from indoor scenes, which is tedious work. In addition, these methods can only extract indoor scene components that observe comments in the training set and cannot discover new components.

Researchers have also detected the indoor scene components by other methods. Stein et al. [31] de-composed the scene into an adjacency-graph of surface patches, where edges in the graph were classified as either convex or concave. Then the locally convex connected sub-graphs were extracted as components of indoor objects. Kaick et al. [32] presented a segmentation method for components with complete and incomplete shapes where the shape was first decomposed into approximate convex parts, then these were merged into consistent components based on a non-local geometric signature. Due to poor connectivity caused by missing parts and outliers of point clouds, the methods are not suitable for detecting indoor scene components from point clouds, especially for components with complex shapes.

We propose a learning-free method that mainly exploits the similarity and spatial proximity of the profiles of slices of indoor scene components. There are some similar slicing-based methods [45–47,49,50] proposed to detect components of individual objects. Differently from these methods, our method mainly detects components of the whole indoor scenes. The segmentation of the indoor scene is more complex than that of the individual objects. By our method, the complex-shaped indoor scene components (from different individual objects) can be effectively extracted.

2.2. The Acquisition of Spatial Relationships

Extracting spatial relationships is crucial for the understanding of indoor scenes. There are many efforts that have been expended on the extraction of spatial relationships from scene images. Muda [33] used region boundaries and region labels to generate annotations describing absolute object positions and also relative positions between pairs of objects

on the base of a domain ontology and spatial information ontology. Aditya et al. [34] presented a general architecture where the generic visual recognition techniques for the image scenes were implemented. Then a mapping between scene categories and inferred scene constituents was collected and implemented to predict relationships between scene constituents. Xu et al. [35] proposed a novel end to end model that solved the scene graph inference problem using standard Recurrent Neural Networks (RNNs) and learnt to iteratively improve predictions on objects and their relationships via message passing. In contrast with the spatial relationships in 2D images, the spatial relationships in the 3D point clouds are far more complex and the extraction of them is challenging.

Existing methods for the acquisition of spatial relationships directly from 3D point clouds can be divided into two categories, i.e., the machine learning-based methods and the knowledge-based methods. The machine learning-based methods mainly predefine the types of spatial relations and then train a classifier to predict the spatial relations. Silberman [38] introduced a principled approach that integrated physical constraints and statistical priors on support relationships to reason spatial semantics such as support from back, support from below, etc. Choi et al. [39] introduced a 3D Geometric Phrase Model (3DGP) which defined a group of object types (e.g., sofa, chair, table, etc.) and their 3D spatial configuration and proposed a latent SVM method to learn the interactions among scene objects. Because spatial relations are complex, it is difficult to establish a fixed parametric model for training. Thereby, the deep learning technology has recently been adopted by some researchers. Wald et al. [40] proposed two PointNet architectures for the extraction of objects and their spatial relationships and exploited a Graph Convolutional Networks to process the acquired object–object relationships. Although the deep learning techniques have shown prospects in the extraction of certain spatial relations, obtaining effective features of the complex spatial relationships is still difficult.

The knowledge-based methods mainly use prior spatial knowledge to infer spatial relationships. Zender et al. [42] used an innate conceptual ontology that defined abstract categories for rooms and objects and how they are related to create conceptual map representations of human-made environments to represent spatial properties of typical indoor environments. Suchan and Bhatt [43] proposed an ontological characterization of human activities to extract commonsense spatio-temporal relations and patterns (e.g., left-of, touching, part-of, during, approaching, etc.) to offer human-centered automated reasoning about embodied spatio-temporal interactions with indoor environments. Ponciano et al. [44] proposed a knowledge system to detect the specific components of indoor objects and interleaved between spatial semantics inference and object recognition some spatial relationships (e.g., around, parallel) that had a close connection with recognition of indoor objects were extracted. In the context of spatial semantics extraction on synthetic data, Kontakis et al. [41] mimicked human spatial cognition and presented a knowledge-based index mechanism for the automated spatial correlation between objects in terms of linguistic predicts. Existing knowledge-based methods mostly handle the spatial relationships between indoor objects. In our work, we use the spatial knowledge to infer the spatial relationships between indoor scene components from 3D point clouds. By introducing a slice strategy, the acquisition of spatial relationships between indoor scene components is facilitated.

3. Overview

Our work consists of two stages, i.e., the detection of indoor components and the spatial relationships inference. The framework is shown in Figure 1.

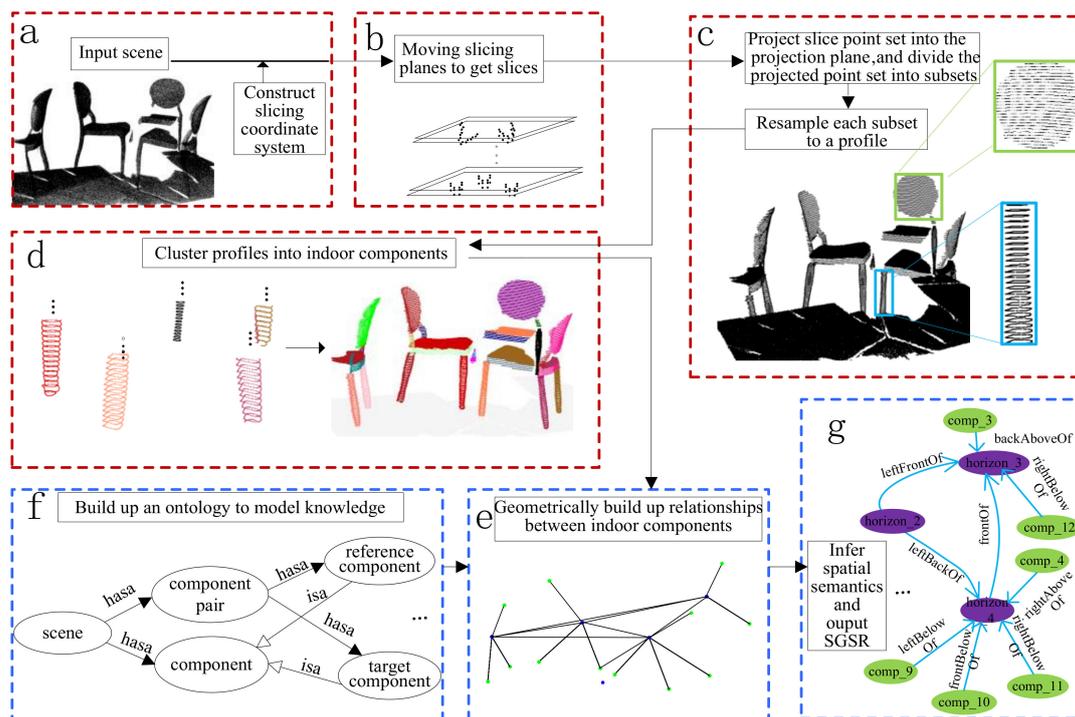


Figure 1. The framework of the proposed method: (a) Input an indoor scene model; (b) Slicing the indoor scene model; (c) Generate the profiles; (d) Cluster profiles into different indoor scene components; (e) Construct the relationship graph; (f) Build up an ontology to model knowledge about indoor scene components; (g) Infer spatial semantics and output SGSR of the input scene model.

(1) Detection of indoor scene components

- (a) Given an indoor scene model, we firstly adopt a simple direction searching strategy to label the ground. Then we construct a slicing coordinate system (see Figure 1a), where the center of the bounding rectangle of the ground is taken as the origin, and the upward normal of the ground is taken as the z-axis. The x - and y -axes are chosen from two arbitrary orthogonal axes on the ground.
- (b) In the slicing coordinate system, from bottom to top, we iteratively slice the indoor scene model using two planes by a step size h in the perpendicular direction to the z-axis (see Figure 1b).
- (c) We project the point set of a slice on a projection plane and divide the projected point set into many subsets. Then each subset is resampled to a profile (see Figure 1c).
- (d) Profiles on different projection planes are clustered into indoor scene components based on the similarity and spatial proximity (see Figure 1c,d).

(2) Spatial relationships inference

- (a) We geometrically build up relationships between indoor scene components (see Figure 1e).
- (b) An ontology is constructed to model the commonsense knowledge about the semantics of spatial relationships between indoor scene components (see Figure 1f). Then the ontology is populated by the geometrically correlated indoor scene components.
- (c) The pair-wise spatial relationships are inferred by SWRL rules. An SGSR of the indoor scene model is output to represent the indoor scene components and their spatial relationships (see Figure 1g).

4. Detection of Indoor Scene Components

4.1. Slicing and Resampling of Indoor Point Clouds

To slice the indoor scene model effectively, the slicing coordinate system is constructed, where the center of the bounding box of the ground is taken as the origin. The upward normal of the ground is selected as the z -axis, and two arbitrary orthogonal axes on the ground are chosen as x -axis and y -axis.

It is observed that most indoor objects are placed upright on the ground. Therefore, the ground can be labeled through a simple direction searching. The specific process is as follows. (1) Compute the Orientation Bounding Boxes (OBBs) of the indoor scene model and obtain outer planes $\Pi_i, i = 0, 1, 2 \dots m$ ($m \leq 5$) that correspond with the ground, the walls, and the ceiling. (2) Filter the points belonging to outer planes and segment the indoor scene model PC into point sets $PC = \cup_i (P_i)$ by a k -nearest-neighbor (KNN) algorithm. (3) We select $\Pi_i, i = 0, 1, 2 \dots m$ ($m \leq 5$) as the ground and roughly regard each point set P_i as an object and generate OBBs from the resulting point sets. Moreover, due to the assumption that most objects are parallel to the ground, we enforce this constraint for the OBB computation—the orientation along the parallel plane of the plane Π_i . (4) If a Π_i has the largest number of OBBs closest to itself, it is identified as the ground.

Motivated by the aim of ensuring that enough geometric features are included in each slice, we characterize the indoor scene slice as an indoor scene section with a thickness of l . The thickness l is computed by $\lambda_d d_{dens}$, d_{dens} denotes the density of the point clouds, λ_d is a density factor. d_{dens} is formulated as the following equation,

$$d_{dens} = \frac{1}{N} \sum_{i=1}^N \frac{1}{K} \left(\sum_{k=1}^K \|p_i - p_k\| \right) \quad (1)$$

where $p_i, i = 1, \dots, N$ denote a point of indoor point clouds, p_k is the k -closest point of p_i . K is set to 6.

The slicing position is initialized at the point that has the minimum z -axis value in the slicing coordinate system. Starting from the initial slicing position (the lower slicing plane is located at the initial position), from bottom to up, we iteratively slice the input indoor point clouds using two slicing planes by a step size h in the slicing direction, as seen in Figure 2.

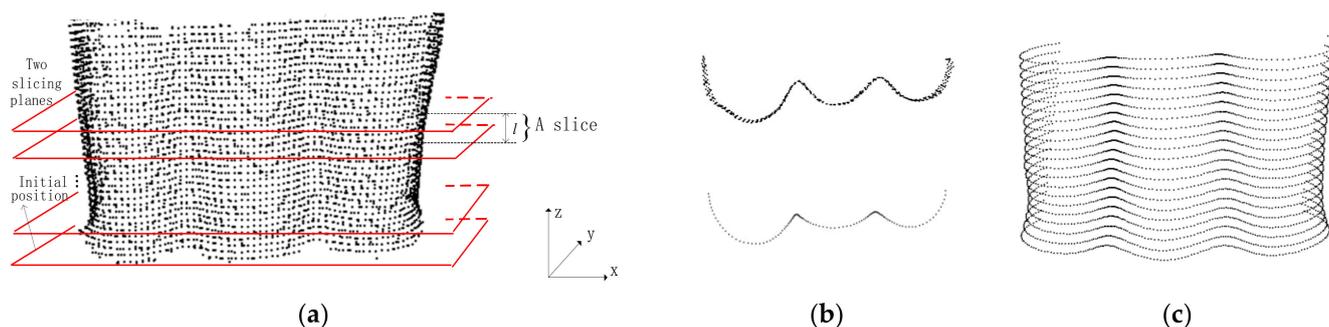


Figure 2. Slicing and resampling of point clouds: (a) Slicing of point clouds; (b) A slice and a profile; (c) Profiles of the point clouds.

For each slice, a plane parallel to the slicing planes and located between the two slicing planes and equidistant from the two slicing planes is defined as the projection plane. On this basis, the point set of each slice is projected to the projection plane by setting the z -axis value of each point to the z -axis value of the intersection point of the projection plane and z -axis.

The projected point set of a slice is first divided into some subsets by the clustering algorithm [51]. Then each subset is thinned using the Moving Least Squire (MLS) method [52], and is thereafter resampled to a profile with an interval d . The size of d is calculated as $d = \frac{1}{N} \sum_{i=1}^N \frac{1}{K} \sum_{k=1}^K \|p_i - p_k\|$, where p_i is a point of indoor point clouds, p_k is

the k -closest point of p_i . K is set to 5. Figure 2b shows one of slices of the object and the resampled point set, i.e., the profile of the slice. Figure 2c shows all the profiles of an indoor object.

Note that some special subsets do not need to be resampled. We divide the minimum bounding box (MBB) of each subset into many sub-rectangles and label the sub-rectangles that include one or more projected points, then count the labeled sub-rectangles and total sub-rectangles. If the ratio of a labeled sub-rectangles number to total sub-rectangles number is bigger than 0.7, the subset does not need to be resampled. More details about dividing MBB into sub-rectangles can be seen in [49]. The special subsets directly constitute a special kind of component of the indoor scene model. We refer to them as horizontal plane components (horizontal planes for short).

To obtain the appropriate value of the density factor λ_d , we performed experiments on indoor scene models with different densities. By using different sampling rates to down-sample the indoor point cloud, point clouds with different densities can be obtained. Given a tabletop scene model (Figure 3a), we chose the original model, the 50% down-sampling model, and the 25% down-sampling model for the experiments. We set λ_d to 0.14, 0.23, and 0.34 for the models. The results are shown in Figure 3b–l). If λ_d is smaller, the slice will be thinner. The thinner the slice, the fewer points on the slice. In a severe case, profiles will fail to be obtained. As can be seen, the profiles are largely missing when λ_d is 0.14 or 0.23. When λ_d is 0.34, good results are achieved. In our work, λ_d was set to 0.34.

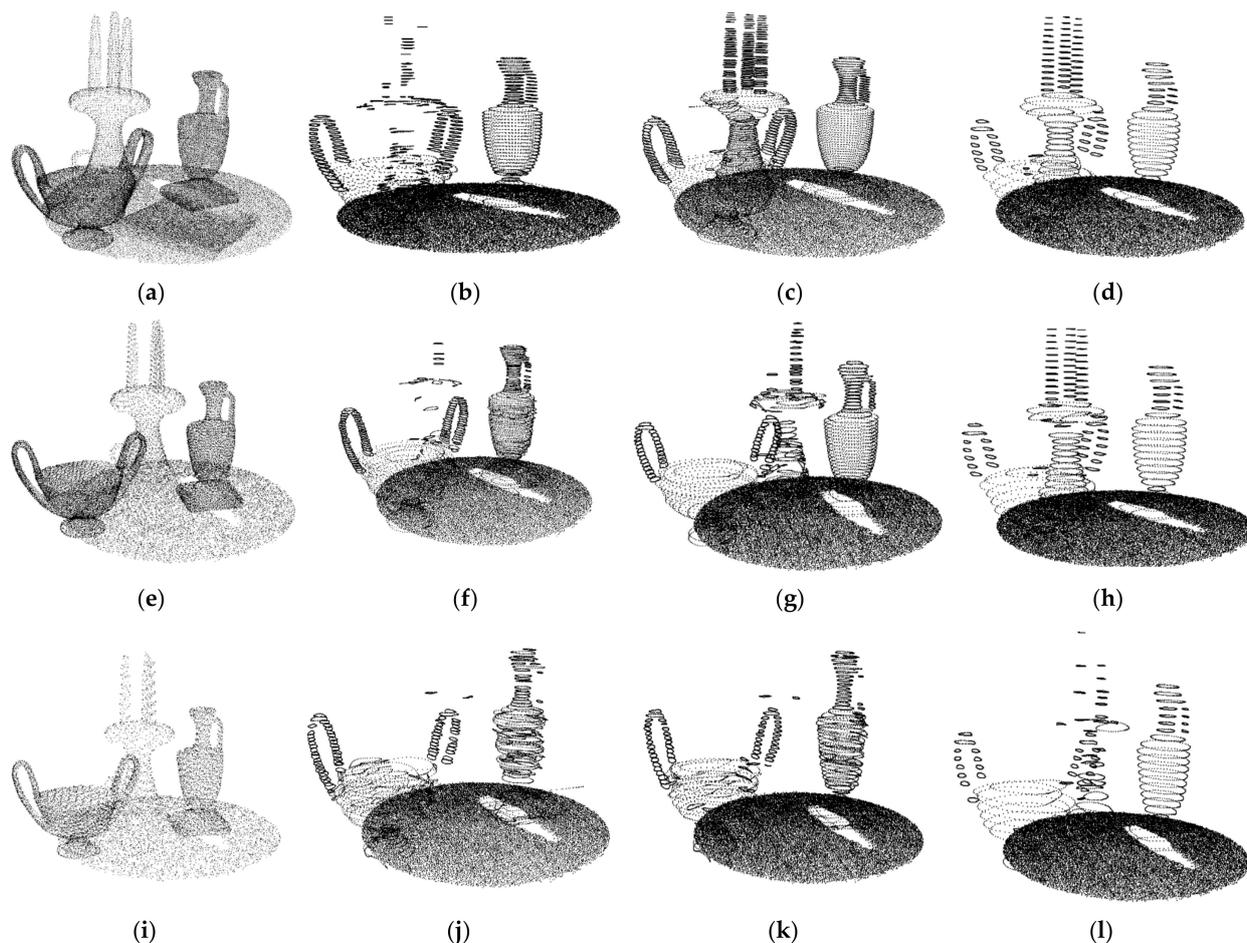


Figure 3. Experimental results of l : (a) Scene 1 (99538 points), $d_{dens} = 1.431$; (b) $\lambda_d = 0.14$, $l = \lambda_d d_{dens}$; (c) $\lambda_d = 0.23$, $l = \lambda_d d_{dens}$; (d) $\lambda_d = 0.34$, $l = \lambda_d d_{dens}$; (e) 50% down-sampling model, $d_{dens} = 1.995$; (f) $\lambda_d = 0.14$, $l = \lambda_d d_{dens}$; (g) $\lambda_d = 0.23$, $l = \lambda_d d_{dens}$; (h) $\lambda_d = 0.34$, $l = \lambda_d d_{dens}$; (i) 25% down-sampling model, $d_{dens} = 2.779$; (j) $\lambda_d = 0.14$, $l = \lambda_d d_{dens}$; (k) $\lambda_d = 0.23$, $l = \lambda_d d_{dens}$; (l) $\lambda_d = 0.34$, $l = \lambda_d d_{dens}$.

h depends on the expected number of slices. h will affect the running time. The smaller the h , the longer the cutting and resampling process will take, as shown in Table 1. Different slicing results of the scene are shown in Figure 4. H was finally set to $1.0 l$.

Table 1. Executing times.

h	$0.4 l$	$0.8 l$	$1.0 l$	$2 l$	$4 l$
times	450 s	285 s	240 s	180 s	126 s

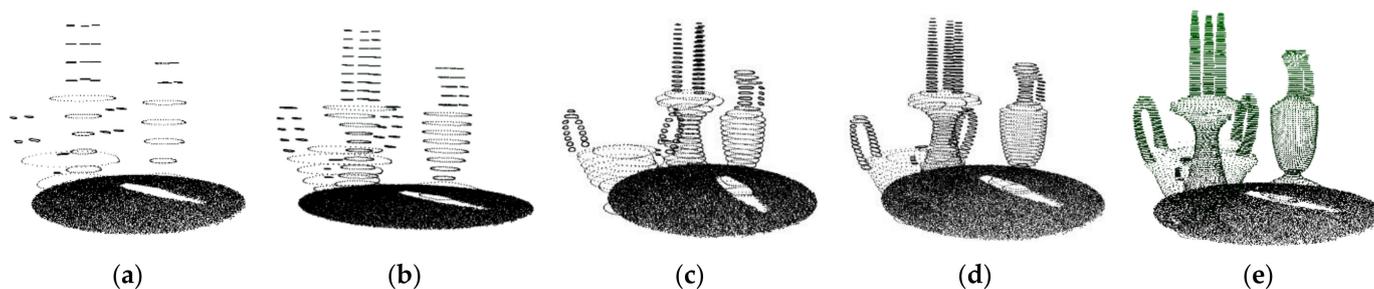


Figure 4. Experimental results of h : (a) $l = 0.34 d_{dens}$, $h = 4 l$; (b) $l = 0.34 d_{dens}$, $h = 2 l$; (c) $l = 0.34 d_{dens}$, $h = 1.0 l$; (d) $l = 0.34 d_{dens}$, $h = 0.8 l$; (e) $l = 0.34 d_{dens}$, $h = 0.4 l$.

4.2. Clustering of Profiles

Let the total resampled point set of indoor scene model be \mathfrak{R} , and let each profile be λ_{ij} (i.e., the j th profile of the i th slice), then $\mathfrak{R} = \left\{ \left\{ \lambda_{ij} \right\}_{j=1}^{M_i} \right\}_{i=1}^N$.

Given two profiles λ_{ij} and λ_{qk} of \mathfrak{R} , their spatial proximity and similarity are evaluated. To judge whether two profiles are adjacent, their MBB s are calculated and denoted as MBB_1 and MBB_2 , respectively. If λ_{ij} and λ_{qk} belong to neighboring slices and MBB_1 and MBB_2 are overlapped, λ_{ij} and λ_{qk} have spatial proximity.

For two profiles with spatial proximity, their similarity is further judged. A similarity measure $D_{sc}(\lambda_{ij}, \lambda_{qk}) + (1 - \min(\frac{MBB_1}{MBB_2}, \frac{MBB_2}{MBB_1}))$ is designed, where $D_{sc}(\lambda_{ij}, \lambda_{qk})$ computes the distance between shape context features [53] of the two profiles, $\min(\frac{MBB_1}{MBB_2}, \frac{MBB_2}{MBB_1})$ is adopted to approximate the scale ratio of the two profiles. If the similarity measurement between λ_{ij} and λ_{qk} is smaller than a threshold δ , the two profile λ_{ij} and λ_{qk} belong to the same component. Starting from the initial profiles, the profile pair in \mathfrak{R} are iteratively clustered into different components, i.e., the profile clustering-based components.

To evaluate the effect of threshold δ on the clustering results, we set δ to 0.38, 0.48, 0.58, respectively, and the clustering results are shown in Figure 5. It can be seen that the smaller δ may result in over-segmentation (see the blue rectangle in Figure 5b), and the bigger δ may result in under-segmentation (see the blue rectangle in Figure 5d). We set δ to 0.48 in our work.

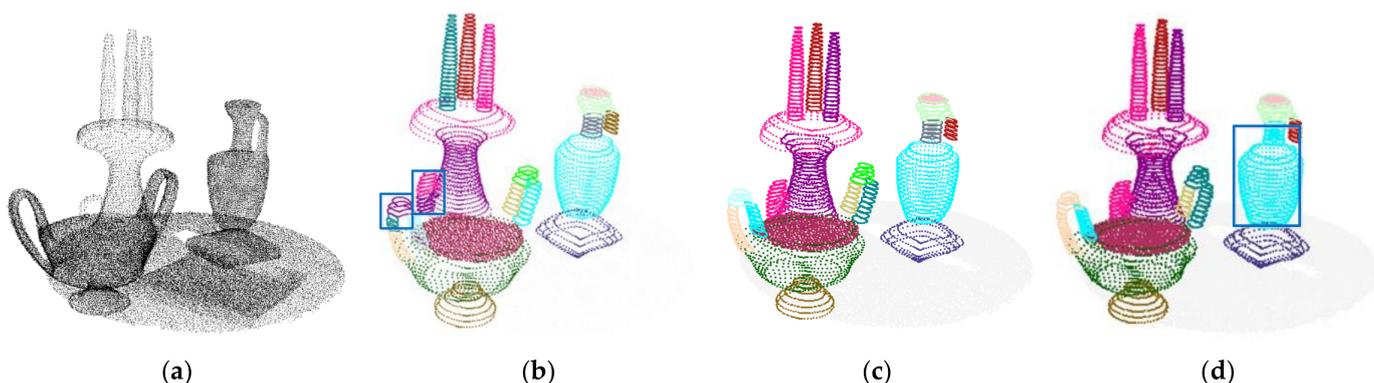


Figure 5. Experimental results of δ : (a) Scene 1; (b) $\delta = 0.38$; (c) $\delta = 0.48$; (d) $\delta = 0.58$.

Note that some complex-shaped components may be over-segmented due to the profiles in some local surfaces of the components (see the blue rectangle in Figure 6a). To solve the problem, we will locally adjust the slicing direction in a way similar to the method in [45] at this local surface. Specifically, given three components, $Comp_1$, $Comp_2$, $Comp_3$, if a profile of $Comp_1$ is respectively overlapped with its neighbor's profiles that belong to $Comp_2$ and $Comp_3$ (see Figure 6a), we label the component set as $\{Comp_1, Comp_2, Comp_3\}_c$, and we reslice the raw points that correspond with $Comp_1$. A rotational slicing direction will iteratively be applied in the raw points (see Figure 6b) until the optimal slicing is found. Then we re-cluster the re-generated profiles and the profiles of $Comp_2$ and profiles of $Comp_3$, and update the clustering results (see Figure 6c) according to the minimum number of components principle.

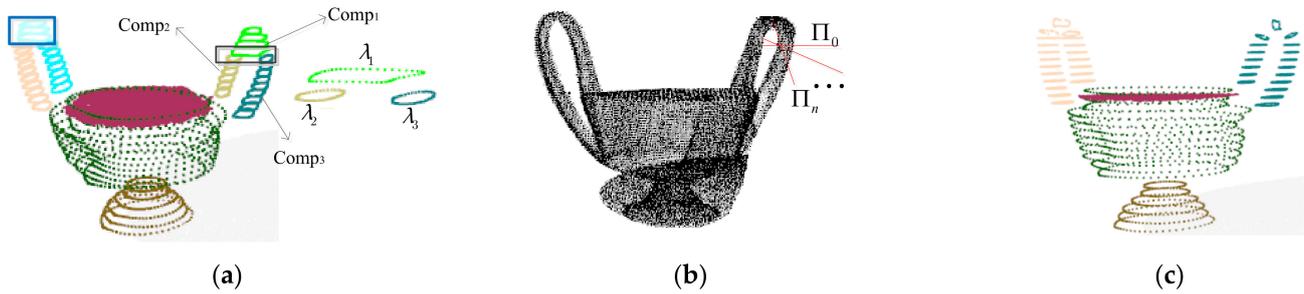


Figure 6. Adjustment of slicing: (a) The initial segmentation result; (b) rotated slicing planes; (c) the final results.

Algorithm 1. Clustering profiles into indoor scene components.

Input: $\mathfrak{R} = \left\{ \left\{ \lambda_{ij} \right\}_{j=1}^{M_i} \right\}_{i=1}^N$, λ_{ij} is the j th profile on i th projection plane, δ

Output: $\{Comp_l\}$

1. $l = 0$;
 2. **for** $i = 1:1:N$
 3. **for** $j = 1:1:M_i$ **do** $Comp_l \leftarrow \phi$;
 4. **if** λ_{ij} is not marked
 5. $\lambda_s \leftarrow \lambda_{ij}$, mark λ_{ij} ; //
 6. $u = i + 1$; **do** $\Gamma \leftarrow \{\lambda_{uv}\}_{v=1}^{M_u}$ & λ_{uv} is not marked; $\Gamma \leftarrow \phi$;
 7. search spatial&similar profile λ_r of λ_s in Γ ;
 8. $Comp_l \leftarrow Comp_l \cup \lambda_r$; mark λ_r ; $\lambda_s \leftarrow \lambda_r$;
 9. $u = u + 1$;
 10. **until** λ_r is not found
 11. $l = l + 1$; **end if**
 12. **end for**
 13. **end for**
 14. Search component set $\{Comp_p, Comp_q, Comp_k\}_c$ from $\{Comp_l\}$, $p, q, k \in [0, l]$,
 15. **for each** $\{Comp_p, Comp_q, Comp_k\}_c$
 16. apply rational-direction slice in raw points that corresponding with $Comp_p$
 17. re-generate profiles;
 18. re-cluster the profiles and profiles of $Comp_q, Comp_k$, and update $\{Comp_l\}$
 19. **end for**
 20. output $\{Comp_l\}$
-

5. Spatial Semantics Inference

5.1. Spatial Relationships

5.1.1. Topological Relationships

Topology is a fundamental aspect of space. There are some popular formalizations of topological relations in 3D space. Region Connection Calculus (RCC) [36] is a popular formalization of topological relationships. It provides occlusion support by considering the

projections of 3D objects in 2D space. In [37], the Dimensionally Extended Nine-Intersection Model (DE-9IM) defined 11 types of topological relationship such as disjoint, meet, contains, covers, inside, covered by, equal, etc., according to the boundary-based intersection pattern between two 3D objects.

In our work, we mainly focus on two kinds of topological relationships between indoor scene components, i.e., connect, adjacent. They are defined on the basis of two geometric distances. The first distance is the minimal distance between two components. It is calculated as $d_1(A, B) = \inf_{p_1 \in A, p_2 \in B} d(p_1, p_2)$, where A and B are components, p_1 and p_2 are two points that belong to A and B , respectively. The second distance is the distance between the centroids of two components. It is calculated as $d_2(A, B) = d(\text{centroid}(A), \text{centroid}(B))$.

Because indoor scene components have different scales, e.g., the components of furniture and the components of tabletop objects, we adopt the following principles for building up topological relationships between two indoor scene components.

(1) We divide the indoor scene components into two categories, i.e., *Class_I* and *Class_II*, according to their scales. Specifically, if the area of a horizontal plane is bigger than the area threshold, or the volume of a profile clustering-based component is bigger than the volume threshold, the component is classified into *Class_I* components. Otherwise, the component is classified into *Class_II* components.

(2) For two *Class_I* components A and B , if $d_1(A, B) < \sigma_1$, they are connected. If $d_1(A, B) > \sigma_1$ and $d_2(A, B) < \sigma_2$, they are adjacent. In our work, σ_1 was set to 0.35, σ_2 was set to 3.

(3) For two *Class_II* components A and B , if $d_1(A, B) < \sigma_1$, they are connected. If $d_1(A, B) > \sigma_1$ and $d_2(A, B) < \sigma_2$, they are adjacent. σ_1 was set to the same value as that of *Class_I* component. σ_2 was set to 0.75.

(4) For a *Class_I* horizontal plane and a *Class_II* component, we will judge whether they are connected or not. The adjacent relationships between them are not considered. Moreover, we only consider the connected relationship between the *Class_I* horizontal plane and the *Class_II* components. The topological relationships between the other *Class_I* components and *Class_II* components are not considered.

5.1.2. Directional Relationships

Directional relations refer to another major category of spatial analysis. The directional relationships mainly describe the relative position of 3D components in a coordinate system. In order to depict the directional relationships between the indoor components finely, we divide 3D space around an indoor scene component into 14 sub-spaces, i.e., above, below, left, right, front, back, left_above_back, left_above_front, etc., as seen in Figure 7.

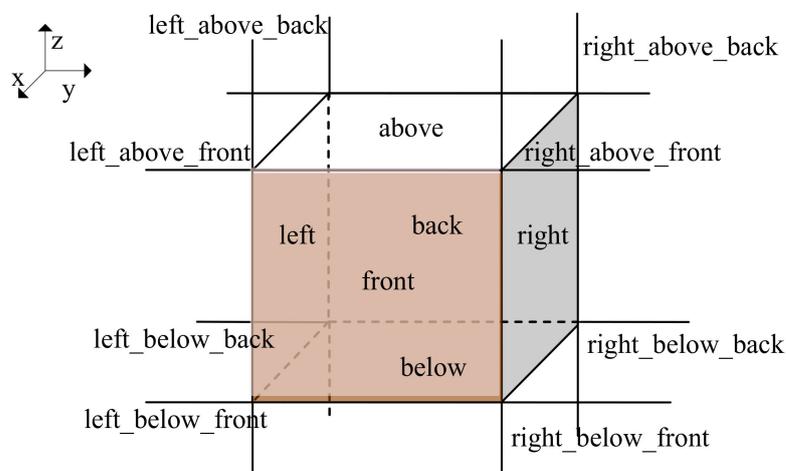


Figure 7. Directional relationships.

Obviously, given a reference component B , when a component A is located in the six subspaces around the reference component, i.e., above, below, left, right, front, and back, there at exists least six corresponding directional relationships between A and B , i.e., leftOf, rightOf, frontOf, backOf, ect. We take the leftOf or rightOf as the examples to illustrate how to decide the directional relationships. The slicing coordinate system is taken as the reference coordinate system. The directional relationship of indoor scene components is determined through the coordinates of vertices of the indoor scene component's MBBs.

We first project their MBB vertices that have maximum and minimum y -axis coordinate value onto the y -axis, as shown in Figure 8. Then we calculate the length of the longest line segment formed by the projected vertices, i.e., l_y . Next, for MBB of A or B , we calculate the distance between the vertices that respectively have the largest and the smallest y -axis coordinate value, i.e., l_a and l_b . Let $l_{y0} = l_a + l_b$, if l_y/l_{y0} is larger than 0.8, and $A_MBB.maxy \geq B_MBB.maxy$, then A is in the right of B (see the blue rectangle A'), as seen in Equation (2). On the contrary, A is in the left of B (see the red rectangle A''), as shown in Equation (3).

$$isRightOf(A, B) \leftarrow l_y/l_{y0} \geq \tau \ \&\& \ A_MBB.maxy \geq B_MBB.maxy \tag{2}$$

$$isLeftOf(A, B) \leftarrow l_y/l_{y0} \geq \tau \ \&\& \ A_MBB.miny < B_MBB.miny \tag{3}$$

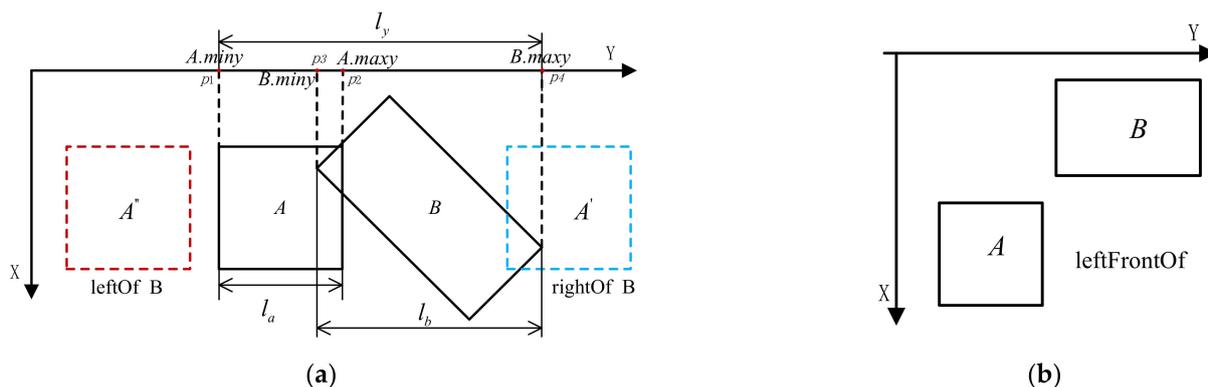


Figure 8. Directional relationships: (a) The leftOf and rightOf relationships; (b) the leftFrontOf relationships.

In some cases, there may be two directional relationships between component B and component A simultaneously. For example, if component A is located at the front-left of component B , as seen in Figure 8b. This two spatial relations are collectively denoted as *isleftFrontOf* (A,B). It depends on the projection of their MBBs on x -axis and y -axis. There also are some similar directional relationships, e.g., *leftFrontOf* (A,B), *rightFrontOf* (A,B), etc.

If component A is located in the eight sub-spaces around the reference component B such as *left_above_back*, *left_above_front*, etc., there will be corresponding relationships *leftFrontAboveOf* (A,B), and *rightFrontOf* (A,B), etc., between them. Similarly, these directional relationships are defined on the x -axis, y -axis, and z -axis coordinates.

5.2. Ontology-Based Spatial Semantics Inference

We built up an ontology to model the common sense knowledge about the semantics of spatial relationships between indoor scene components, as seen in Figure 9a. The ontology consists of a number of concepts arranged hierarchically. The root concept is a scene concept with two sub-concepts, i.e., component and component pair. The component pair concept includes two sub-concepts, i.e., the reference component and the target component. The hierarchical spatial relationship between indoor scene components is shown in Figure 9b. The properties of the ontology concepts are enumerated in Table 2.

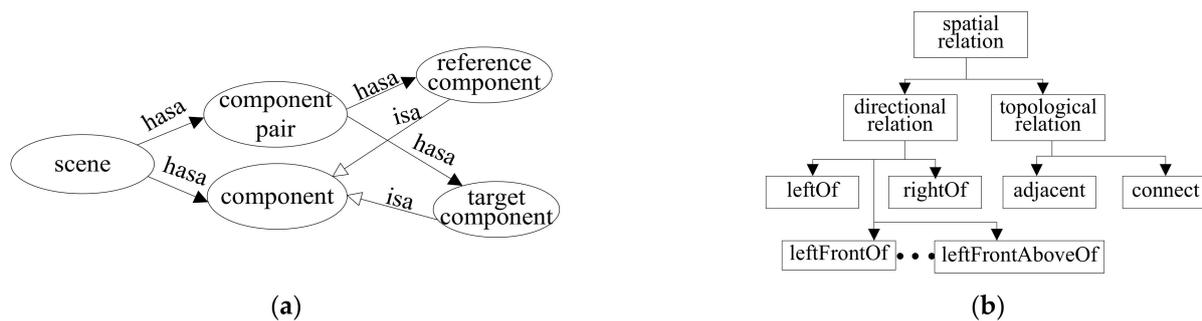


Figure 9. The scene ontology: (a) The scene ontology; (b) the spatial relationships.

Table 2. Properties of ontology concepts.

Properties of Component Concept	Meaning	Properties of Component Pair Concept	Meaning
hasType	horizontal plane or profile clustering-based component	hasReferenceComp	a relation starting from a component
hasArea	area of the horizontal plane	hasTargetComp	a relation target at a component
hasVolume	volume of MBB of profile clustering-based component		
hasMBB	MBB of the component		

We firstly geometrically built relationships between indoor scene components on the base of two geometric distance d_1 and d_2 and obtain component pairs. Then the ontology concepts are populated by a related component. For each component pair instance, the horizontal plane is preferred as the reference component, and the profile clustering-based component is preferred as the target component. If both components are horizontal planes or the profile clustering-based components, any one of them is instantiated as the reference component, and the other one is instantiated as the target component.

We mainly adopt Semantic Web Rule Language (SWRL) to regulate rules for inferring spatial semantics. For instance, the following assert the topological relationships and spatial relationships between two components, respectively.

$isAdjacent(?A,?B) \rightarrow isComPair(?comPair) \wedge hasReferenceComp(?comPair, ?A) \wedge hasTargetComp (?comPair, ?B) \wedge Distance1 (?A, ?B, ?dis_1) \wedge swrlb: greaterThan(?dis_1, \sigma_1) \wedge Distance2(?A,?B, ?dis_2) \wedge swrlb: greaterThan(?dis_2, \sigma_2)$

$isLeftOf(?A,?B) \rightarrow isComPair(?comPair) \wedge hasReferenceComp(?comPair, ?A) \wedge hasTargetComp (?comPair, ?B) \wedge hasMBB (?A,?A_MBB) \wedge hasMBB(?B,?B_MBB) \wedge ?Length1(?A_MBB,?B_MBB,?ly) \wedge Length2(?A_MBB,?B_MBB,?lyo) \wedge MinY(?A_MBB,? A_MBB_miny) \wedge MinY(?B_MBB,?B_MBB_miny) \wedge swrlb: greaterThan(?ly, \tau lyo) \wedge swrlb: smallerThan(?A_MBB_miny,? B_MBB_miny)$

We adopt a graph to represent the indoor scene components and spatial relationships as seen in Figure 10a, where the light green nodes represent the *Class_II* profile clustering-based components, the light purple nodes represent the *Class_II* horizontal plane, the dark purple nodes represent the *Class_I* horizontal plane, a pair of nodes connected by edges represent two geometrically related components. The component in each component pair is taken as the node of SGSR. The inferred spatial relationships are added into SGSR as edges as shown in Figure 10b, where the blue edges denote the adjacent relationships, and the yellow edges connect relationships. To limit redundancy, the adjacent relationships between the components that share a horizontal plane have been filtered. The directional relationships whose reference component and target component have been exchanged each other have also not been shown.

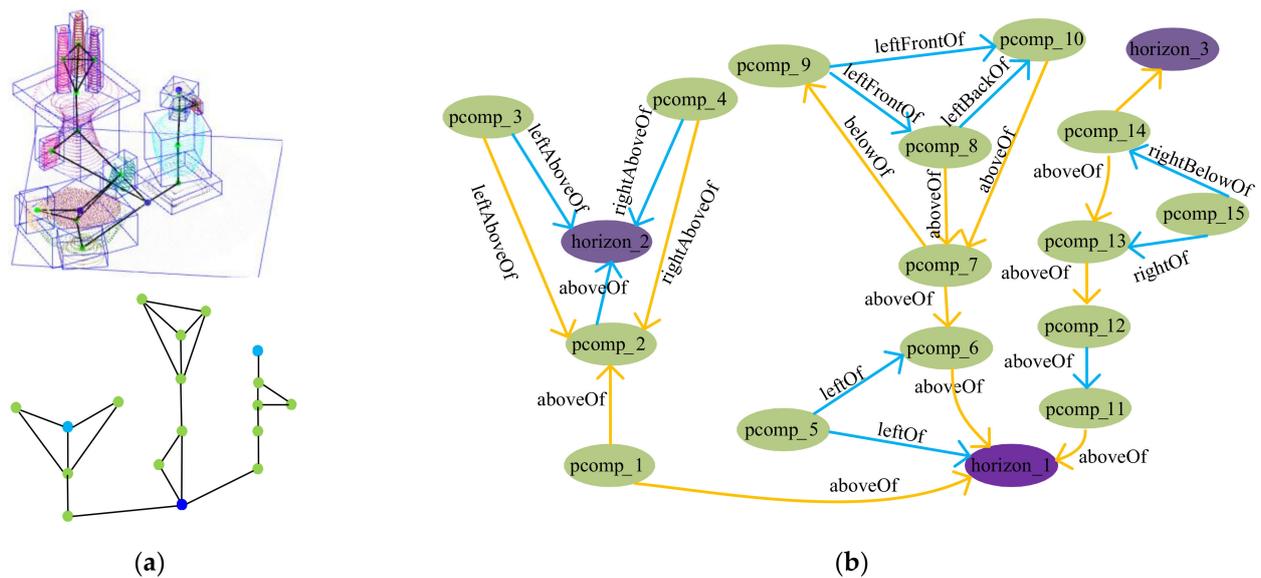


Figure 10. SGSR of a scene: (a) The tabletop scene and the relationships graph of scene; (b) SGSR of scene.

6. Experiments

6.1. Evaluation of Indoor Scene Components Segmentation

6.1.1. Experiments of Indoor Scene Components Detection

To evaluate the effectiveness of the proposed method, we ran an extensive set of experiments on some indoor scene models. The proposed algorithm was programmed with VC++ and OpenGL for display and rendering. All of the experiments in this paper were run on a PC with Intel CoreI2, CPU2.80GHz, 2G memory. We evaluated our method on ETH [54], TUM [55] and dataset [56]. We empirically set $\lambda_d = 0.34$, $h = 1.0 \times l$, $\delta = 0.48$ for the experimental scenes. ETH is a high-quality point cloud dataset containing 18 office scenes. It mainly includes indoor objects such as chair, desk, keyboard, monitor, mouse, cup, cabinet, lamp, sofa, pillow and so on. Clutter and occlusion were presented in the dataset. In order to show the scenes clearly, we removed the walls, ceilings and ground from the scenes. The qualitative experimental results are shown in Figure 11. They show that most of the components were detected correctly. Besides, some objects in the scenes were cuboid-like or cylinder-like (e.g., cabinets, boxes, and bottles). Each of them was simply structured and should have been an individual component of the indoor scene. However, considering that our proposed method can segment out the plane parallel to the slicing planes from each object, these objects were labeled as two components in the ground truth, i.e., the body and the cover (i.e., a plane parallel to the slicing planes). Although it is inconsistent with the general ground truth (i.e., these objects are individual components), this segmentation (i.e., label the body and the cover of the object) is still meaningful.

The dataset [56] included different kinds of scenes such as living rooms, office rooms, meeting rooms. Figure 12a–d respectively shows the detected indoor components of living room, office, lounge, and meeting room. It can be seen that most indoor scene components have been detected successfully, which is expected from the results presented in the previous section.

TUM dataset is a low-quality RGBD dataset and its scenes include persons. Generally speaking, persons have more complex shapes than furniture. Through the TUM dataset, the effectiveness of the proposed method for complex shaped indoor components was evaluated. Figure 13a–c respectively shows the detected indoor scene components of three kinds of scenes that including person with different poses. It can be seen that the walls, computer screens, table tops, most body parts of persons have been detected effectively. Moreover, the TUM dataset is made up of low-quality RGBD data, which increased the difficulty of detection of components. Due to the adoption of profile features instead of

point features, the general object and human separation of our method was still very good. The executing time of our method on the three datasets are shown in Table 3.



Figure 11. Experiment results of ETH scenes, different colors denote the components of indoor scenes: (a) Input scenes; (b) Top views of components of scenes; (c) Front views of components of scenes.

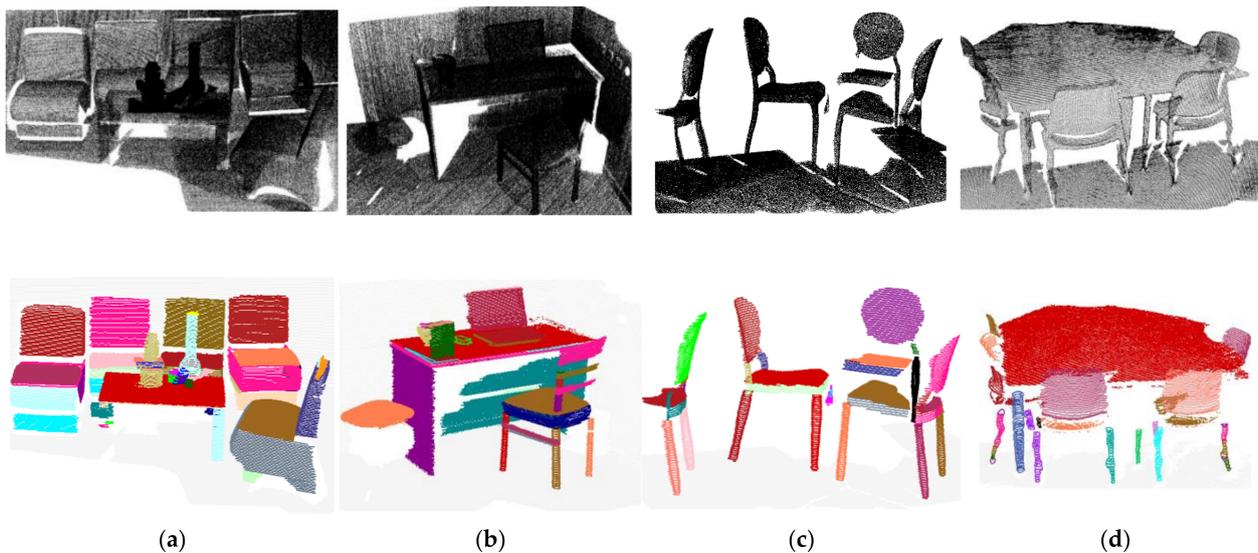


Figure 12. Experimental results of scenes of dataset [56], the first row is the input scene model, the second row is the indoor scene components detection result: (a) Living room; (b) Office; (c) Lounge; (d) Meeting room.

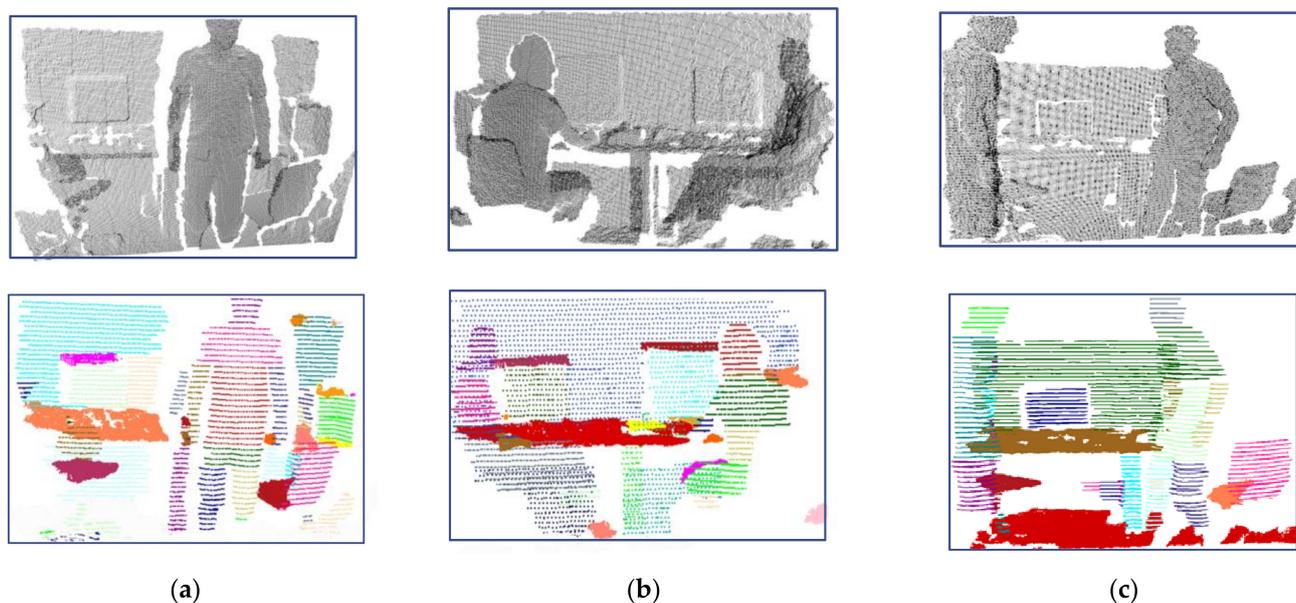


Figure 13. Experimental results of TUM scenes, the first row is the input scene model, the second row is the indoor scene components detection result: (a) Scene with a standing person; (b) Scene with two sitting person; (c) Scene with two standing person.

Table 3. Mean executing times on dataset.

Dataset	ETH	Dataset [56]	TUM
Mean time	4300 s	2105 s	1100 s

A few special-shaped components of objects were over-segmented, e.g., the chair back in Figure 14a. However, the main part of the chair back (the red rectangle) was detected and preserved. A limitation of our method is the determination of slicing direction. We adopted a fixed slicing direction that was orthogonal with the normal direction of the ground. If some objects are not placed on the ground with an up-right posture, some components will not be detected correctly, as shown in Figure 14c.

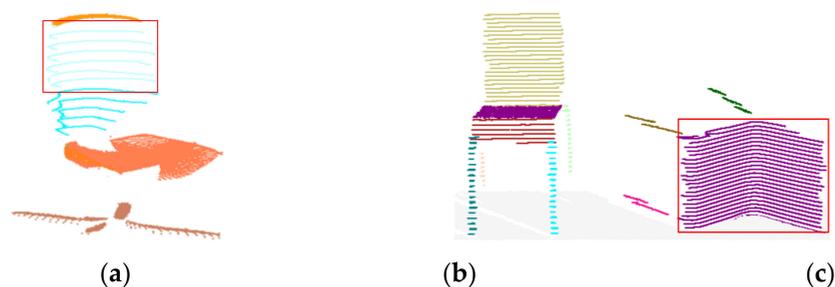


Figure 14. Limitations: (a) Objects with special shaped components; (b) Objects with normal posture; (c) Objects with abnormal posture.

6.1.2. Comparison of the Methods

To demonstrate the advantage of the proposed method, we compared our method with RANSAC [16], a local convexity-based method (LCB) [31], a CRF-based method [27] and PointNet++ [28] on the tabletop scene and the meeting room. The results are shown in Figure 15. It can be seen that the complex-shaped indoor scene components such as the bowl handle and some chairs and legs were not detected exactly by LCB and RANSAC. With the help of machine-learning technology, CRF achieved better results than LCB and

RANSAC. However, CRF still failed to detect some components due to the fact that the optimal CRF model parameters are difficult to obtain. The semantic segmentation results of PointNet++ are shown in Figure 15d, where the components semantic categories are labeled. The indoor scene components with same semantic categories are further separated by a classification network and the instance components are shown in Figure 15e. A few of the components were not detected accurately by PointNet++ because of the wrong semantic label of points.

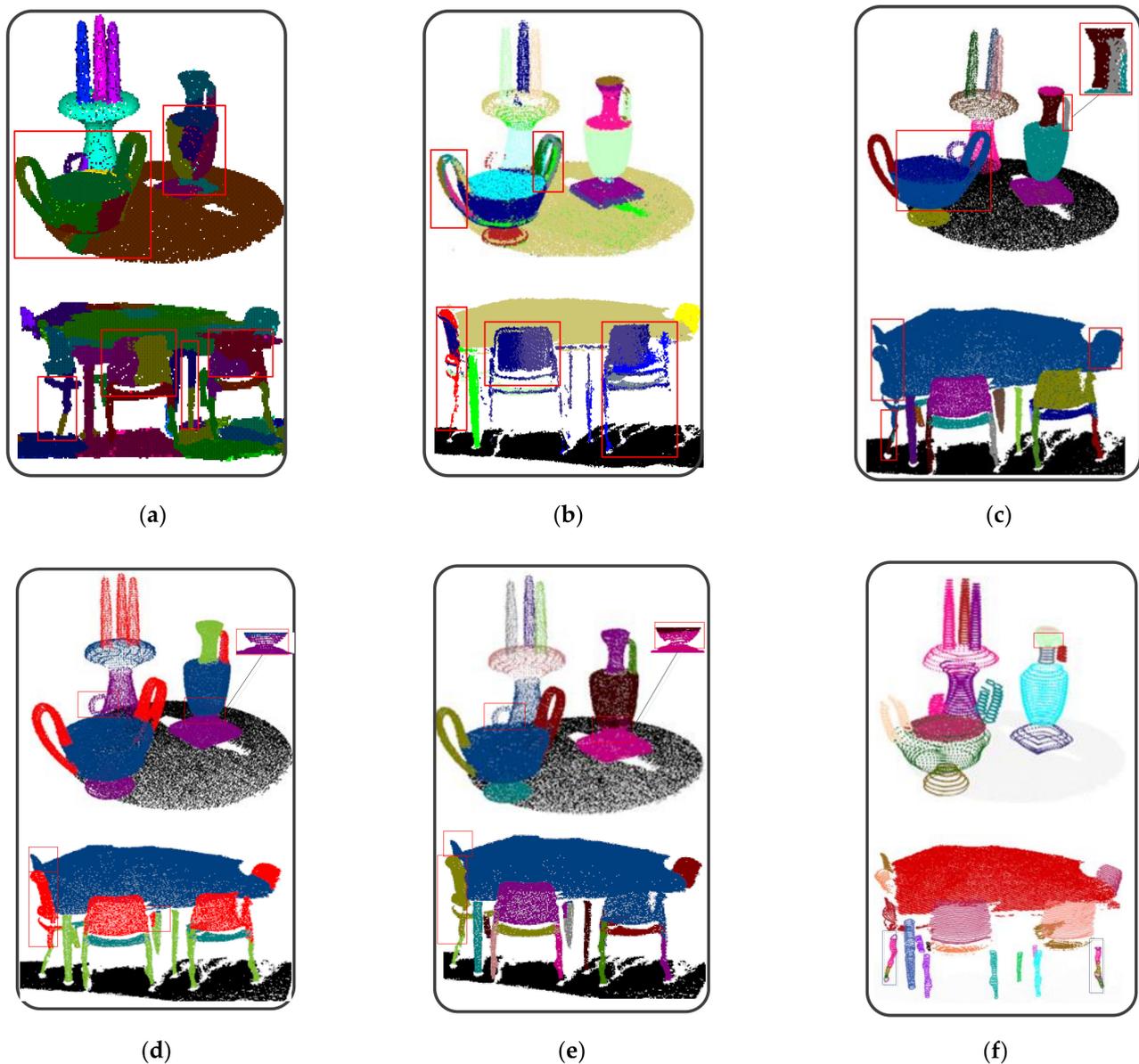


Figure 15. Components of the indoor scene models: (a) LCB; (b) RANSAC; (c) CRF; (d) Semantic segmentation result of PointNet++; (e) Instance segmentation result of PointNet++; (f) The proposed method.

The quantitative results of Figure 15 are shown in Table 4, where the totally detected components numbers (TN) and the correctly detected components numbers (RN) are counted. A detected indoor scene component was considered to be correctly detected if its IoU overlap ratio with the *MBB* of the ground truth components was larger than a threshold. IoU is the volume of the 3D intersection of the *MBBs*, divided by the volume of their 3D union. Here, the threshold overlap ratio was set to 0.7. The ground truth indoor

scene components of the scenes were manually labeled, and the ground truth numbers (GN) of the tabletop scene and meeting room were 17 and 20, respectively. Table 4 shows more of the components of the scenes correctly detected by the proposed method.

Table 4. The quantitative results of Figure 15.

	LCB		RANSAC		CRF		PointNet++		Our Method	
	TN	RN	TN	RN	TN	RN	TN	RN	TN	RN
Tabletop scene	23	9	28	16	17	15	14	14	18	17
Meeting room	33	14	20	9	20	15	19	18	21	15

The quantitative results of the experiments are shown in Table 5. The extraction ratio and error ratio of indoor scene components was introduced to evaluate the methods. The extraction ratio was expressed as $ratio_extra = \frac{RN}{GN}$. Moreover, the error extraction ratio was formulated as $ratio_error = \frac{TN-RN}{TN}$. It shows that the proposed method had a higher extraction ratio than LCB and RANSAC. In comparison with CRF and PointNet++, the proposed method achieved a higher extraction ratio of the indoor scene components. However, because the proposed method may have generated more segments, its error extraction may have also been higher than CRF and PointNet++. In comparison with LCB, CRF, and PointNet++, the proposed method had a lower IoU overlap ratio with the ground truth. This was due to the over-segmentation of some special shapes.

Table 5. The extraction ratio of indoor scene components and topological relationships.

	Components			Error Ratio of Components			Mean IoU Overlap	Topo-Relationships
	EHT	Dataset [56]	TUM	EHT	Dataset [56]	TUM		
LCB	0.60	0.55	0.49	0.56	0.64	0.66	0.92	-
RANSAC	0.66	0.65	0.70	0.43	0.50	0.41	0.86	-
CRF	0.86	0.84	-	0.17	0.18	-	0.90	-
Baseline [57]	-	-	-	-	-	-	-	0.72
Our method	0.86	0.87	0.88	0.23	0.20	0.18	0.87	0.83
PointNet++	0.87	0.84	-	0.16	0.15	-	0.89	-

6.2. Evaluation of Spatial Semantics Acquisition

Here, we report the 3D spatial semantics inference results of our method. We represent the component and the spatial relationships of the meeting room by SGSR, as seen in Figure 16a. We compared our method against a relationship prediction baseline inspired by baseline [57]. The baseline extracted indoor components from point clouds on the base of 3D primitive shapes approximation and built up the topological relationships (e.g., adjacent, parallel, and orthogonal, etc.), as seen in Figure 16b. We represented each spatial relationship as a triplet ($Comp_i, relationship, Com_j$) and compared the triplets of SGSR with the ground truth triplets. If $Comp_i$ and Com_j of a triplet, respectively, had an IoU overlap ratio of 0.7 or higher with that of a ground truth triplet, and the relationship of the triplet was also the same with that of the ground truth triplet, the triplet of SGSR was considered as correctly predicted. The extraction rate was defined as the number of the correctly predict triplets against the number of the ground truth triplets. The triplets of generated inspired by baseline [57] were also compared with the ground truth in the same way. The extraction ratio of the proposed method and the baseline are shown in Table 5. It shows that SGSR represented the structure of indoor scenes more exactly.

Data Availability Statement: Publicly available datasets were analyzed in this study. This data can be found here: <https://www.ifi.uzh.ch/en/vmml/research/datasets.html>; <https://3d.bk.tudelft.nl/liangliang/publications.html>; <https://vision.in.tum.de/data/datasets/rgbd-dataset/download> (accessed on 14 December 2021).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Chen, X.-T.; Li, Y.; Fan, J.-H. RGAM: A novel network architecture for 3D point cloud semantic segmentation in indoor scenes. *Inf. Sci.* **2021**, *571*, 87–103. [CrossRef]
2. Hu, S.-M.; Cai, J.-X.; Lai, Y.-K. Semantic labeling and instance segmentation of 3D point clouds using patch context analysis and multiscale processing. *IEEE Trans. Vis. Comput. Graph.* **2018**, *26*, 2485–2498. [CrossRef] [PubMed]
3. Lin, Y. Hierarchical Co-Segmentation of 3D Point Clouds for Indoor Scene. In Proceedings of the 2017 International Conference on Systems, Signals and Image Processing (IWSSIP), Poznan, Poland, 22–24 May 2017; pp. 1–5.
4. Chen, K.; Lai, Y.-K.; Wu, Y.-X. Automatic semantic modeling of indoor scenes from low-quality RGB-D data using contextual information. *ACM Trans. Graph.* **2014**, *33*, 1–12.
5. Chen, J.; Kira, Z.; Cho, Y.K. Deep learning approach to point cloud scene understanding for automated scan to 3D reconstruction. *J. Comput. Civ. Eng.* **2019**, *33*, 04019027.
6. Wang, J.; Xie, Q.; Xu, Y. Cluttered indoor scene modeling via functional part-guided graph matching. *Comput. Aided Geom. Des.* **2016**, *43*, 82–94. [CrossRef]
7. Schnabel, R.; Wessel, R.; Wahl, R. Shape recognition in 3d point-clouds. In Proceedings of the 16th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision, Bonn, Germany, 7 January 2008; pp. 65–72.
8. Hao, W.; Wang, Y. Structure-based object detection from scene point clouds. *Neurocomputing* **2016**, *191*, 148–160. [CrossRef]
9. Benko, P.; Kós, G.; Várady, T. Constrained fitting in reverse engineering. *Comput. Aided Geom. Des.* **2002**, *19*, 173–205. [CrossRef]
10. Li, Y.; Wu, X.; Chrysathou, Y. Globfit: Consistently fitting primitives by discovering global relations. *ACM Transact. Graph.* **2011**, *30*, 1–12.
11. Li, J.; Xu, K.; Chaudhuri, S. Grass: Generative recursive autoencoders for shape structures. *ACM Trans. Graph.* **2017**, *36*, 52–65. [CrossRef]
12. Hashemifar, Z.S.; Lee, K.W.; Napp, N. Consistent cuboid detection for semantic mapping. In Proceedings of the 2017 IEEE 11th International Conference on Semantic Computing (ICSC), San Diego, CA, USA, 30 January–1 February 2017; pp. 526–531.
13. Vosselman, G.; Gorte, B.G.; Sithole, G. Recognising structure in laser scanner point clouds. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2004**, *46*, 33–38.
14. Guo, S.; Pridmore, T.; Kong, Y. An improved Hough transform voting scheme utilizing surround suppression. *Pattern Recognit. Lett.* **2009**, *30*, 1241–1252. [CrossRef]
15. Schnabel, R.; Wahl, R.; Klein, R. Efficient RANSAC for point-cloud shape detection. *Comput. Graph. Forum* **2007**, *26*, 214–226. [CrossRef]
16. Papazov, C.; Burschka, D. An efficient ransac for 3d object recognition in noisy and occluded scenes. In Proceedings of the Asian Conference on Computer Vision, Queenstown, New Zealand, 8–12 November 2010; pp. 135–148.
17. Papon, J.; Abramov, A.; Schoeler, M. Voxel Cloud Connectivity Segmentation—Supervoxels for Point Clouds. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 2027–2034.
18. Xu, Y.; Hoegner, L.; Tuttas, S. Voxel-and graph-based point cloud segmentation of 3d scenes using perceptual grouping laws. In Proceedings of the ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume IV-1/W1, 2017 ISPRS Hannover Workshop: HRIGI 17-CMRT 17-ISA 17-EuroCOW 17, Hannover, Germany, 6–9 June 2017.
19. Lin, Y.; Wang, C.; Zhai, D. Toward better boundary preserved supervoxel segmentation for 3d point clouds. *ISPRS J. Photogramm. Remote Sens.* **2018**, *14*, 39–47. [CrossRef]
20. Wolf, D.; Prankl, J.; Vincze, M. Fast semantic segmentation of 3D point clouds using a dense CRF with learned parameters. In Proceedings of the 2015 IEEE International conference on robotics and automation (ICRA), Seattle, WA, USA, 26–30 May 2015; pp. 4867–4873.
21. Hausman, K.; Balint-Benczedi, F.; Pangercic, D. Tracking-based interactive segmentation of textureless objects. In Proceedings of the 2013 IEEE International Conference on Robotics and Automation, Karlsruhe, Germany, 6–10 May 2013; pp. 1122–1129.
22. Qi, C.R.; Su, H.; Mo, K. Pointnet: Deep learning on point sets for 3d classification and segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 652–660.
23. Yi, L.; Huang, H.; Liu, D. Deep part induction from articulated object pairs. *ACM Trans. Graph.* **2018**, *37*, 209–214. [CrossRef]
24. Yu, X.; Savarese, S. Estimating the aspect layout of object categories. In Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012; pp. 3410–3417.
25. Te, G.; Hu, W.; Zheng, A. Rgcnn: Regularized graph cnn for point cloud segmentation. In Proceedings of the 26th ACM International Conference on Multimedia, Seoul, Korea, 22–26 October 2018; pp. 746–754.
26. Yi, L.; Kim, V.G.; Ceylan, D. A scalable active framework for region annotation in 3D shape collections. *ACM Trans. Graph.* **2016**, *35*, 1–12. [CrossRef]

27. Xiao, L.; Casas, J.R.; Pargas, M. 3D Point Cloud Segmentation Using a Fully Connected Conditional Random Field. In Proceedings of the 25th European Signal Processing Conference (EUSIPCO), Kos, Greece, 28 August–2 September 2017; pp. 1–6.
28. Qi, C.R.; Yi, L.; Su, H. PointNet++: Deep hierarchical feature learning on point sets in a metric space. *arXiv* **2017**, arXiv:1706.02413.
29. Bokhovkin, A.; Ishimtsev, V.; Bogomolov, E. Towards Part-Based Understanding of RGB-D Scans. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 7484–7494.
30. Balado, J.; Díaz-Vilariño, L.; Arias, P. Point Clouds to Direct Indoor Pedestrian Pathfinding. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2019**, XLII-2/W13, 753–759. [[CrossRef](#)]
31. Stein, C.S.; Schoeler, M.; Papon, J. Object partitioning using local convexity. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 304–311.
32. Kaick, O.V.; Fish, N.; Kleiman, Y. Shape segmentation by approximate convexity analysis. *ACM Trans. Graph.* **2014**, *34*, 1–11. [[CrossRef](#)]
33. Muda, Z.; Lewis, P.H.; Payne, T.R. Enhanced image annotations based on spatial information extraction and ontologies. In Proceedings of the 2009 IEEE International Conference on Signal and Image Processing Applications, Kuala Lumpur, Malaysia, 18–19 November 2009; pp. 173–178.
34. Aditya, S.; Yang, Y.; Baral, C. Image understanding using vision and reasoning through scene description graph. *Comput. Vis. Image Underst.* **2018**, *173*, 33–45. [[CrossRef](#)]
35. Xu, D.; Zhu, Y.; Choy, C.B. Scene graph generation by iterative message passing. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 5410–5419.
36. Albath, J.; Leopold, J.L.; Sabharwal, C.L. RCC-3D: Qualitative Spatial Reasoning in 3D. In Proceedings of the 23rd International Conference on Computer Applications in Industry and Engineering, International Society for Computers and Their Applications, Cordoba Spain, 1–4 June 2010; pp. 74–79.
37. Strobl, C. Dimensionally extended nine-intersection model (de-9im). In *Encyclopedia of GIS*; Springer: New York, NY, USA, 2008; pp. 240–245.
38. Silberman, N.; Hoiem, D.; Kohli, P. Indoor segmentation and support inference from rgb-d images. In Proceedings of the European Conference on Computer Vision, Florence, Italy, 7–13 October 2012; pp. 746–760.
39. Choi, W.; Chao, Y.-W.; Pantofaru, C. Understanding indoor scenes using 3d geometric phrases. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 33–40.
40. Wald, J.; Dhama, H.; Navab, N. Learning 3d semantic scene graphs from 3d indoor reconstructions. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 3961–3970.
41. Kontakis, K.; Malamos, A.G.; Steiakaki, M. Object identification based on the automated extraction of spatial semantics from web3d scenes. *Ann. Emerg. Technol. Comput.* **2018**, *2*, 1–10. [[CrossRef](#)]
42. Zender, H.; Mozos, O.M.; Jensfelt, P. Conceptual spatial representations for indoor mobile robots. *Robot. Auton. Syst.* **2008**, *56*, 493–502. [[CrossRef](#)]
43. Suchan, J.; Bhatt, M. Commonsense scene semantics for cognitive robotics: Towards grounding embodied visuo-locomotive interactions. In Proceedings of the IEEE International Conference on Computer Vision Workshops, Venice, Italy, 22–29 October 2017; pp. 742–750.
44. Ponciano, J.-J.; Trémeau, A.; Boochs, F. Automatic detection of objects in 3D point clouds based on exclusively semantic guided processes. *ISPRS Int. J. Geo-Inf.* **2019**, *8*, 442. [[CrossRef](#)]
45. Wang, Y.; Zhang, H.; Wang, N. Rotational-guided optimal cutting-plane extraction from point cloud. *Multimed. Tools Appl.* **2020**, *79*, 7135–7157. [[CrossRef](#)]
46. Goyal, M.; Murugappan, S.; Piya, C. Towards locally and globally shape-aware reverse 3D modeling. *Comput. Aided Des.* **2012**, *44*, 537–553. [[CrossRef](#)]
47. Zhou, Y.; Yin, K.; Huang, H. Generalized cylinder decomposition. *ACM Trans. Graph.* **2015**, *36*, 1–14. [[CrossRef](#)]
48. Wang, Y.; Asafi, S.; Van Kaick, O. Active co-analysis of a set of shapes. *ACM Trans. Graph.* **2012**, *31*, 165–174. [[CrossRef](#)]
49. Wang, Y.; Zhang, H.; Zhao, Y. Three-dimensional reconstruction method of Tang Dynasty building based on point clouds. *Opt. Eng.* **2015**, *54*, 123111. [[CrossRef](#)]
50. Yin, K.; Huang, H.; Zhang, H. Morfit: Interactive Surface Reconstruction from Incomplete Point Clouds with Curve-Driven Topology and Geometry Control. *ACM Trans. Graph.* **2014**, *33*, 1–12. [[CrossRef](#)]
51. Viswanath, P.; Pinkesh, R. l-dbscan: A fast hybrid density based clustering method. In Proceedings of the 18th International Conference on Pattern Recognition (ICPR'06), Venice, Italy, 22–29 October 2006; pp. 912–915.
52. Lee, I.-K. Curve reconstruction from unorganized points. *Comput. Aided Geom. Des.* **2000**, *17*, 161–177. [[CrossRef](#)]
53. Belongie, S.; Malik, J.; Puzicha, J. Shape matching and object recognition using shape contexts. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *24*, 509–522. [[CrossRef](#)]
54. Mattausch, O.; Panozzo, D.; Sorkine-Hornung, O. Object detection and classification from large-scale cluttered indoor scans. *Comput. Graph. Forum J. Eur. Assoc. Comput. Graph.* **2014**, *33*, 11–21. [[CrossRef](#)]
55. Strum, J.; Engelhard, N.; Endres, F. A benchmark for the evaluation of RGB-D SLAM Systems. In Proceedings of the International Conference on Intelligent Robot Systems (IROS), IEEE Robotics and Automation Society, Vilamoura-Algarve, Portugal, 7–12 October 2012; pp. 537–580.

-
56. Nan, L.; Xie, K.; Sharf, A. A search-classify approach for cluttered indoor scene understanding. *ACM Trans. Graph.* **2012**, *31*, 1–10. [[CrossRef](#)]
 57. Hueting, M.; Monzpart, A.; Mellado, N. MCGraph: Multi-criterion representation for scene understanding. In *SIGGRAPH Asia 2014 Indoor Scene Understanding Where Graphics Meets Vision*; ACM: New York, NY, USA, 2014; pp. 1–9.