

Room Partitioning in Complex Environments by Supervoxel Segmentation and Anchor Pixel Linking

Feng Li , WenZhong Shi , and YangJie Sun

Abstract—This article presents an innovative method to automatically partition rooms in cluttered environments, which is a critical task for indoor reconstruction, navigation, as well as scene understanding. It involves two main phases, starting with the operations of morphological erosion and feature analysis on the projected supervoxels to highlight gaps and remove narrow passages. This is followed by a refinement phase, where the continuous and clean wall boundaries are connected in the occupancy evidence map under the constraint of orientation information. Finally, the individualization of rooms is achieved by inversely propagating the segmented result in image back to point cloud. Experimental results demonstrate that the proposed method outperforms mainstream approaches, particularly in challenging scenarios characterized by heavy occlusion, curved walls, multiple ceiling heights, or long corridors.

Index Terms—Indoor space, pixel linking, point cloud, room partitioning, supervoxel segmentation.

I. INTRODUCTION

THE rapid advancements in scanning technology have significantly enhanced the efficiency and scope of capturing indoor scenes, yielding massive point cloud data that intuitively reflect the architectural details [1]. However, the sheer volume of the data and complexity of the environment pose considerable challenges in the effective utilization of the original point cloud [2]. Considering that indoor space is commonly composed of room units with varying size and functional use, a natural strategy is to reduce the amount of data by dividing the building into separated rooms in advance. The room partitioning task not only fosters a deeper understanding of the indoor layout, but also plays a pivotal role in subsequent applications. In the context of indoor reconstruction, focusing on the analysis of a discrete room circumvents interference from extraneous points, thereby improving the geometric accuracy as well as semantic

content of the resultant model [3], [4]. And for indoor navigation, uncovering the spatial topological relationships between rooms is instrumental in devising suitable pathways [5], [6]. Thus, room partitioning stands as a significant topic of research within the fields of architecture, robotics, and spatial analysis [7], [8], [9].

Existing room partitioning methods are broadly classified into two categories, one of which is closely related to the segmentation of occupancy map in the domain of mobile robotics [10], [11], [12]. The division basis for this type of methods is primarily the narrow passages and gaps between rooms in two-dimensional (2D) images. A significant drawback of them resides in the neglect of 3D spatial information, rendering it impossible to cut off the tightly connected rooms just from the planar view [13]. One reliable strategy is to introduce wall information [14], [15], but it remains a formidable challenge to ensure the accuracy and completeness of the extracted walls in cluttered indoor environments. The second type of methods tends to process the point cloud directly and imposes the distribution of solid boundaries or the similarity between subspaces as the primary criteria for room partitioning [16], [17]. However, such methods may require the assistance from additional information, such as the number of rooms, device location, and scanning trajectory [18], [19], [20]. Moreover, these methods are generally limited by regular structural assumptions and incapable of addressing special scenarios where walls are curved or room heights are inconsistent.

In addition to the challenges posed by the intricate indoor structures, the presence of unavoidable noise, outliers, and missing data in the point cloud are most likely to result in incomplete walls and unclear ceiling gaps, which further exacerbate the difficulty of room partitioning [21], [22]. In order to overcome the limitations of existing methods and cope with complex scenarios, this article proposes an automated method for room partitioning from indoor point clouds. The core idea is to outline room boundaries by identifying the structural elements that separate the space, such as walls or doorways. The simplified workflow starts with the application of morphological erosion operations and geometric feature analysis on the projected supervoxels to initially sever the narrow passages that connect rooms. Subsequently, the locally maximal anchor pixels are used to link the wall boundaries under the constraints of directional information, thereby completing the refined partitioning of the rooms. Finally, the segmentation results on the image are inversely propagated back to the point cloud to achieve the individualization of the

Received 18 November 2024; revised 29 March 2025 and 23 May 2025; accepted 27 June 2025. Date of publication 7 July 2025; date of current version 1 August 2025. This work was supported by the Otto Poon Charitable Foundation Smart Cities Research Institute, the Hong Kong Polytechnic University, under Grant CD03. (Corresponding author: WenZhong Shi.)

Feng Li is with the School of History, Culture and Tourism, Fuyang Normal University, Fuyang 236037, China, and also with the School of Environment and Spatial Informatics, China University of Mining and Technology, Xuzhou 221000, China (e-mail: 07152665@cumt.edu.cn).

WenZhong Shi and YangJie Sun are with the Department of Land Surveying and Geo-Informatics, The Hong Kong Polytechnic University, Hong Kong (e-mail: john.wz.shi@polyu.edu.hk; yangjie.sun@polyu.edu.hk).

Digital Object Identifier 10.1109/JSTARS.2025.3586490

room point cloud. The main contributions of this article are threefold.

- 1) A room partitioning method is proposed to handle complex indoor environments. It is not limited to the Manhattan world assumption and does not require additional information.
- 2) The supervoxel segmentation and morphological operations are combined to remove narrow regions and candidate doorways.
- 3) A novel pixel linking strategy that consists of sequential linking, intermittent linking, and jump linking is devised to obtain continuous and clean wall boundaries.

II. RELATED WORK

Current room partitioning methods can be roughly divided into two categories according to the nature of the operation objects. The first category is intimately tied to the utilization of 2D occupancy grid maps, which is herein referred to as indirect partitioning. Conversely, the other category exhibits a stronger preference for the manipulation of 3D point clouds, thus designated as direct partitioning.

A. Indirect Room Partitioning

The introduction of image processing techniques has enabled the segmentation of occupancy grid maps into independent entities. Bormann et al. [23] provide a survey on four popular segmentation algorithms, namely Voronoi graph-based, feature-based, morphology-based, as well as distance transform-based. To effectively apply these methods to the partitioning task of room point cloud, it is necessary to generate the occupancy map by projecting a certain range of points beforehand. The Voronoi graph-based methods generate independent rooms by merging the subdivided spatial units through heuristic rules and topological relations. Ambrus et al. [24] leverage the Voronoi graph to obtain virtual viewpoints from the projection points of the ceiling. The points within a fixed radius from each viewpoint are assigned with a specific label. However, the simple result needs further processing and is prone to oversegmentation. The feature-based methods devise a classifier to associate local features with room labels, but are rarely used in room point cloud partitioning because the performance of the trained classifier degrades significantly in new environments. The morphology-based methods cut off the connections between adjacent room regions by iteratively applying an erosion operation to the occupancy map [25]. Jung et al. [26] and [27] apply the morphological method on a binary image that is generated from the point cloud near the ceiling to preliminarily partition the interior space. Then, the actual extent of individual rooms is determined by closing the opening of skeletonized walls. The disadvantage of it is that undersegmentation problem may occur when the center region of the wall is incorrectly filled with noise points. Yang et al. [28] introduce the wall information to assist the segmentation of tightly connected rooms in the occupancy map. The morphological algorithm is adopted to partition the free-space evidence raster that has been pruned by the wall point evidence into meaningful regions. However, it may obtain incorrect rooms

when the detected walls are disturbed by noise. Wu et al. [29] use cloth simulation filtering algorithm instead of horizontal slicing to extract the ceiling points from the building with multiple room heights. These points are then projected into regular grids, and the connected regions are eroded to separate rooms. However, this method may erroneously remove part of the point cloud near the boundaries. The distance transform-based methods operate by calculating the distance from each occupied raster to the nearest boundary and regarding the position of the raster with the local maxima value as the room center [30]. Tang et al. [31] employ the watershed algorithm on the optimized distance transform map to decompose the space into different functional units. Nevertheless, the final results heavily depend on the correctness of the extracted structure primitives. Martens and Blankenbach [32] utilize distance transformation on the mask images generated jointly by the interior free space and wall grids. Then independent room centers are automatically filtered using the Otsu thresholding method. However, it performs poorly when the walls are extensively obscured or the room size is small.

B. Direct Room Partitioning

The direct methods to room partitioning exhibit little correlation with 2D occupancy maps, and typically revolve around the distribution of walls or doors as well as the similarity among subspaces to segregate rooms [33]. Armeni et al. [34] adopt a template matching algorithm to detect two adjacent peaks formed by a thick wall from the point density histogram and then slice the point cloud along the two dominant axes of the building. But this method is limited to conventional rectangular rooms. Bobkov et al. [35] propose a room partitioning method based on unsupervised clustering. The point cloud is first voxelized and the potential field value is calculated for each free voxel. Then combining the information of visibility, spatial distance, and maximum potential field, a clustering is performed to get room labels. This method does not require the knowledge of device location, but it is difficult to handle buildings with narrow corridors. Elseiczy et al. [36] present a room partitioning method that relies on the sampled trajectory data. Potential locations of doorways are first identified by using point clouds within a fixed radius of the trajectory points. Then the trajectory points are divided into segments and the label value of them are passed to the corresponding subset of the point cloud with the assistance of the temporal attributes. However, due to the presence of openings, the point cloud collected at the same location may be distributed across multiple rooms. Ochmann et al. [37] consider that points with high mutual visibility are more likely to belong to the same room. The ray casting is applied on the planar patches instead of all the point cloud to generate the visibility graph [38]. The Markov clustering algorithm is then applied on the graph to partition rooms. The shortcomings are computational efficiency and over segmentation problems. Cui et al. [39] propose a room partitioning method based on the visibility analysis of trajectory data. First, the visible point cloud of each sampled trajectory point that are limited by doors are calculated. Since the visible point cloud of adjacent trajectory

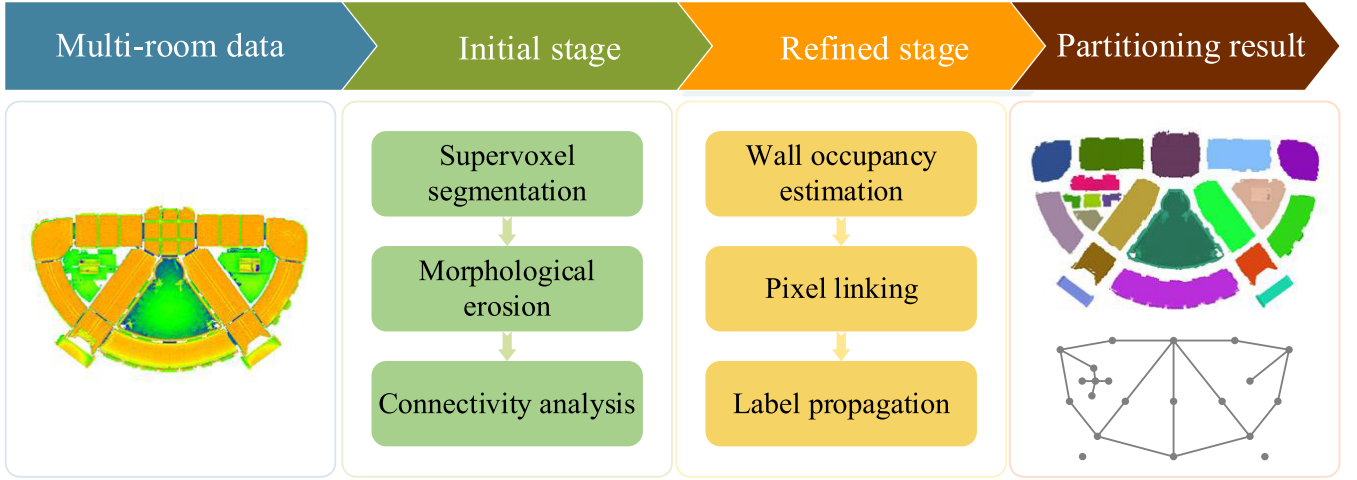


Fig. 1. Overview of the proposed room partitioning method.

points in the same room has a certain overlapping rate, the associated point cloud can be divided into individual rooms accordingly. The limitation of this method is the need to ensure that trajectory data are available. Yang et al. [40] present a room partitioning method that allows for complex buildings with cross-floor spaces or nested rooms. First, the distance between each voxel and the obstacle is calculated using an efficient data structure. Subsequently, the indoor space is packed with a number of spheres that overlap as small to generate room seed regions. But the performance of this method is not good in cluttered or large-scale environments. Liu et al. [41] perform room partitioning through the simulation of the scanning process. A random point in the free space is selected and emits the light ray uniformly in the clockwise direction. Then a virtual enclosing region is formed by extending the walls hit by the ray and marked as a room region if the proportion of the perimeter occupied by real walls is greater than a threshold. Nevertheless, the method is susceptible to the interference from indoor objects, which hinder the propagation of the light ray.

III. METHODOLOGY

The proposed method is designed to partition rooms in complex indoor environments, with its workflow illustrated in Fig. 1. It takes unstructured point clouds as input, which may be captured by various devices such as laser scanners or RGBD cameras. During the initial partitioning phase, the method employs local feature calculation and supervoxel segmentation to divide the input point cloud into a series of compact subsets with better preserved boundary. Then morphological erosion is applied to the projected supervoxels on a horizontal plane, aiming to highlight the gaps between rooms. Simultaneously, narrow doorways are identified and isolated by geometric properties and adjacency relations. In the refinement phase, diverse information including ceiling height, voxel distribution, and point density are combined to build the wall occupancy evidence map, from which the locally maximal pixels are selected and marked as anchor points. Then, under the directional constraint of pixels,

continuous and clean room boundaries are constructed by three kinds of growth criteria, namely sequential linking, intermittent linking, and jump linking. Finally, by wavefront propagation and inverse mapping, the label values are transferred from 2D image to 3D point cloud.

A. Initial Phase

The original point cloud is first downsampled using voxel grids to reduce redundant data while preserving similar spatial resolution across various datasets. Subsequently, the statistical filter is employed to eliminate outlier points, which may be introduced by device noise or environment factors. The sparse outliers are identified by computing the average distance from each point to its k nearest neighbors and comparing it with a predefined threshold. Wherein the value of the threshold depends on the statistical properties of the average distance, namely the mean μ and the standard deviation σ

$$d_i = \frac{1}{k} \sum_{j=1}^k \|p_j - p_i\| \quad (1)$$

$$\text{is_outlier}(p_i) = d_i > \mu + 3\sigma. \quad (2)$$

The next step is to calculate the local geometric features of each point. A covariance matrix M is constructed from the coordinates of neighboring points, and then decomposed using principal component analysis. The eigenvector $\vec{v}_i$ associated with the smallest eigenvalue $\lambda_{\min}$ is regarded as the normal vector of current point, and the surface curvature Cur at the point equals the ratio of the smallest eigenvalue to the sum of all eigenvalues

$$M = \frac{1}{k} \sum_{j=1}^k \left(p_j - \frac{\sum_{j=1}^k p_j}{k} \right) \left(p_j - \frac{\sum_{j=1}^k p_j}{k} \right)^T \quad (3)$$

$$M \cdot \vec{v}_i = \lambda_i \cdot \vec{v}_i, i \in \{1, 2, 3\} \quad (4)$$

$$\text{Cur} = \frac{\lambda_{\min}}{\sum_{i=1}^3 \lambda_i}. \quad (5)$$

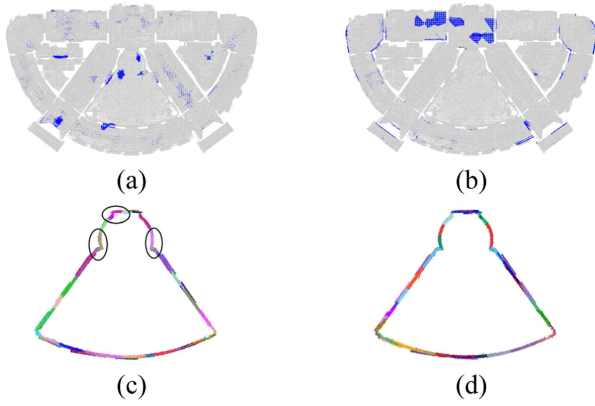


Fig. 2. Comparison results of supervoxel segmentation: (a) and (b) preferentially merged point sets in the original and improved method, and (c) and (d) generated wall supervoxels by the original and improved method.

1) *Supervoxel Segmentation*: Following that, the point cloud is segmented into multiple supervoxels that exhibit internal consistency and distinct boundaries. The method based on fusion and exchange minimization is proposed by Lin et al. [42], which formalizes supervoxel segmentation as a subset selection problem. In order to enhance its segmentation performance in complex indoor scenes, two improvements are made in the article. First, during the merging process, the traversal initiates from the point with the smallest curvature instead of randomly selected point. This strategy is motivated by the fact that points with smaller curvature are typically located in flat regions, and starting the merge operation from these points makes the generated supervoxels adhere more closely to walls, ceilings, or floors. In contrast, the arbitrarily selected points, which may correspond to corners or edges of architectural structures, potentially lead to erroneous aggregation of the points from different components. Second, during the computation of feature distances, in addition to spatial proximity and normal vector difference, maximum curvature is incorporated as a new metric to enhance the robustness against noise. Points that exhibit close spatial coordinates have similar normal vectors and maintain continuous surface which are more likely to be categorized within the same subset. Integrating these three aspects, the specific formulation for computing the feature distance is delineated as follows:

$$D(i, j) = \frac{\omega_s \|p_i - p_j\|}{SR} + \omega_n \left(1 - \left| \vec{n}_i \cdot \vec{n}_j \right| \right) + \omega_c \max(c_i, c_j) \quad (6)$$

where p_i and p_j represent the coordinates of two adjacent points, $\vec{n}_i$ and $\vec{n}_j$ denote the unit normal vectors of them, and c_i and c_j signify their surface curvature values. The resolution of supervoxel is denoted as SR , while ω_s , ω_n , and ω_c are the weighting coefficients used to adjust the importance of the three factors. Fig. 2 provides a visual comparison of the segmentation results before and after the improvement. It can be seen that the original method may start merging points from irregular objects such as decorations or furniture, and probably form supervoxels that cross the actual boundary. Conversely, the improved method

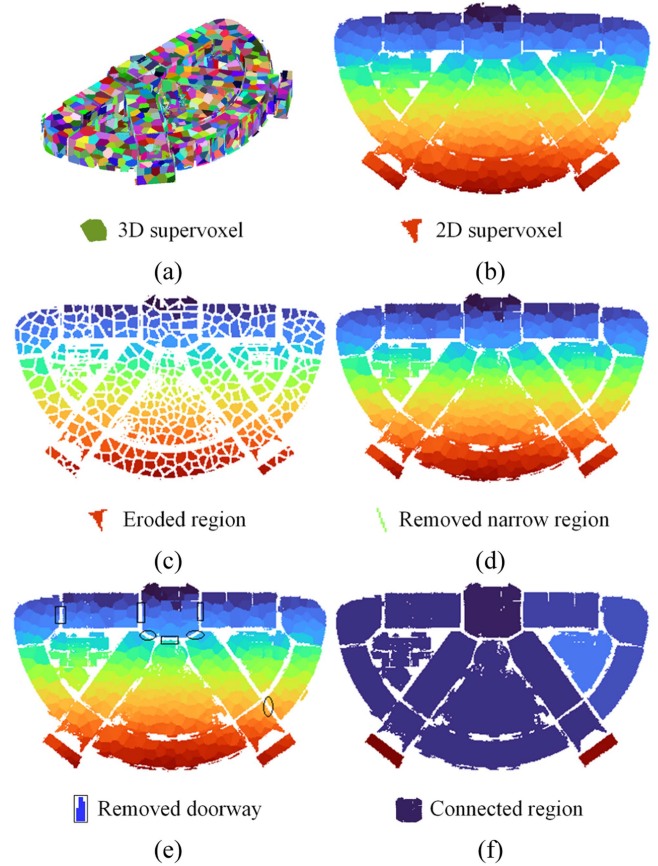


Fig. 3. Outcomes of the initial room partitioning: (a) generated supervoxels, (b) projected regions, (c) eroded image, (d) narrow region removal, (e) doorway removal, and (f) connected regions.

tends to initiate from the points on planar structures and perform better at locations with high surface variation.

The outcomes of initial partitioning are depicted in Fig. 3. First, the supervoxels with insufficient interior points are discarded, as they usually correspond to isolated noisy data. Then, the supervoxels are projected onto a 2D horizontal plane in order to reduce data dimensionality and pave the way for introducing the image processing techniques. Each point of the supervoxel is traversed and assigned to a grid cell in accordance with its planar coordinates. The value of each grid is associated with the supervoxel label of the highest point within it. The grid is then converted to a grayscale image, and its values are mapped into a contiguous integer sequence. As shown in Fig. 3(b), along with the increase of gray value, the rendering color transitions from blue to green and then to red. A collection of pixels with identical grayscale value constitutes a projected region, which serves as a simplified 2D representation of supervoxels.

2) *Morphological Erosion*: Afterwards, morphological erosion is applied to the grayscale image to eliminate noise and narrow structures. Some room partitioning methods disconnect rooms by iteratively eroding edge pixels, but are prone to insufficient or excessive erosion. As opposed to individual pixels, the projected region of the supervoxel is taken as the basic unit for erosion in this article. These projected regions additionally carry

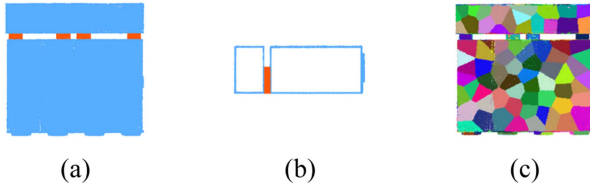


Fig. 4. Geometric features of doorways: (a) spatial extent, (b) height difference, and (c) adjacent regions.

some information from 3D space, the larger regions are closely related to horizontal structures such as ceilings or floors, while smaller regions mostly correspond to walls or noisy objects. The removal of narrow projected regions significantly enhances the clarity of room boundaries. Specifically, a rectangular structural element SE slides over the grayscale image, if any pixel within the extent of this element has a different value from the center pixel, it indicates that the pixel is situated on the edge of a projected region and thus subject to erosion. This process can be mathematically formalized as follows:

$$I_e(x, y) = \begin{cases} 0, & \text{if } \exists (i, j) \in SE, I(x+i, y+j) \neq I(x, y) \\ I(x, y), & \text{otherwise} \end{cases} \quad (7)$$

where $I(x, y)$ and $I_e(x, y)$ represent the pixel value at a certain location in the original image and the eroded image, respectively. As illustrated in Fig. 3(d), narrower regions have disappeared, while the remaining regions still have some pixels after erosion and are entirely preserved. It is noteworthy that in some buildings the sides of crossbeams may also generate narrow regions during projection, but they should be retained to ensure the integrity of the room.

Subsequently, potential doorways are identified from the projected regions of the supervoxels. The incapability of scanning rays to penetrate solid walls results in gaps between adjacent rooms. However, the existence of doorways may tightly connect these point clouds. Therefore, cutting off the doorway can act as an important basis for room partitioning. In the horizontal direction, the spatial extent of a doorway is much smaller than the rooms it connects. And in the vertical direction, the elevation of a doorway is above the floor yet below the surrounding ceiling. As shown in Fig. 4, these geometric features are harnessed to select the projected regions associated with the doorways. If there are at least two independent sets of regions that are not only positioned higher than current region but also larger in area, they typically correspond to rooms on both sides of the passage. In such cases, current region is inferred as a potential doorway and cut off from the image by modifying the values of interior pixels to zero. The black boxes in Fig. 3(e) reveal that the originally connected rooms may be separated after the detected doorways are removed.

3) *Connectivity Analysis*: Finally, the updated grayscale image is divided into multiple independent parts via connectivity analysis. As shown in Fig. 3(f), each part is composed of many

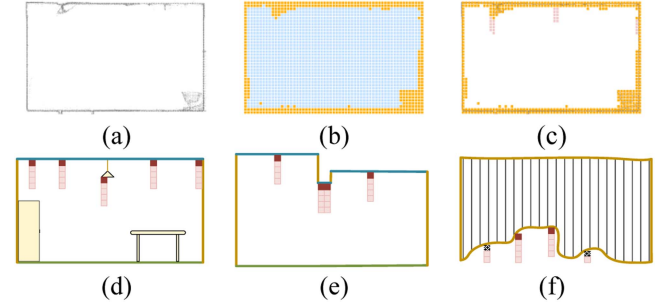


Fig. 5. Accessible height calculation: (a) vertical slice of point cloud, (b) state of voxels, (c) free space, (d) interior of room, (e) doorway, and (f) wall.

2D supervoxels and rendered by a unique color. Upon the completion of morphological erosion and doorway disconnection, the building entity may undergo fragmentation and room boundaries become more prominent. Each connected part comprises a set of spatially continuous and nonempty pixels, which can be effectively identified and labeled through neighborhood search. However, owing to the complexity of indoor environments and the presence of noise data, walls and doorways may be ambiguous, causing many rooms still to remain tightly interconnected. Thus, the aforementioned steps merely achieve an initial partitioning of rooms and require subsequent refinement processes.

B. Refined Phase

Building upon the initial partitioning results, the refinement stage identifies and enhances the wall boundaries by the utilization of point cloud analysis and image processing, thereby ensuring that the final outcome closely aligns with indoor layout. Nonetheless, the accurate and complete extraction of wall structures from unorganized point clouds poses a considerable challenge, particularly in buildings with heavy occlusions and cluttered objects. Therefore, instead of directly segmenting wall surfaces or fitting wall lines, this article generates an occupancy evidence map of the walls by integrating the state of 3D voxels and density of points in 2D grids. Under directional constraints, continuous room boundaries are then extracted by three linking criteria.

1) *Wall Occupancy Estimation*: First, from the perspective of indoor space, the accessible height is derived from the spatial distribution of the point clouds. This attribute is correlated with contiguous and sufficient unoccupied voxels within each vertical grid pillar, exhibiting higher values in the interior of the room and smaller values at the connection area, as illustrated in Fig. 5. During the specific calculation process, vertical supervoxels and ceiling supervoxels are merged into a single point cloud, which is then voxelized in 3D space. As shown in Fig. 5(b), each voxel is assigned with a state of occupied or empty by checking the presence of points within it. To improve insertion and query efficiency, the octree structure is employed to organize and manage the voxel grids. For each grid pillar, a vertical traversal of empty voxels commences from the ceiling and proceeds downward in a sequential manner. The traversal stops when a free space

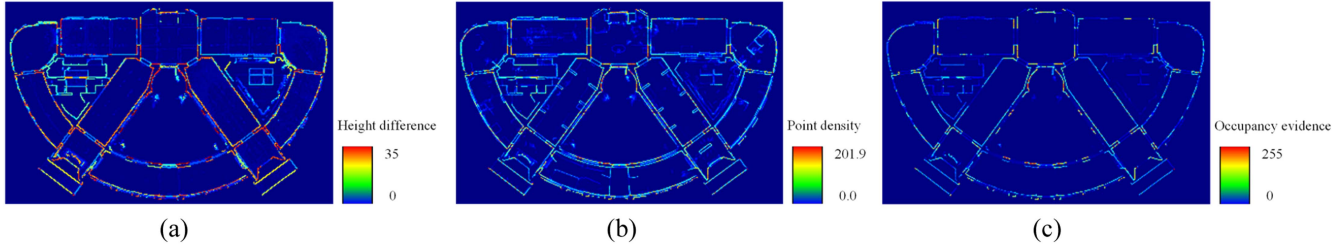


Fig. 6. Construction of wall occupancy evidence map: (a) height difference map, (b) point density map, and (c) wall occupancy evidence map.

with a size exceeding the threshold is detected. Otherwise, it continues searching down from the occupied voxel until enough consecutive empty voxels are found or the floor is reached, as shown in Fig. 5(c). The height of the uppermost empty voxel within the identified free space is recorded and treated as a quantitative value for the accessible height of current grid pillar. Fig. 5(d), (e), and (f) illustrate the accessible heights in typical regions such as the interior of rooms, doorways, and walls, respectively. This attribute not only deepens the understanding of indoor space, but also suffers less in environments with clutter and occlusion. But it focuses more on indoor free space and fails to intuitively highlight the wall space. For this reason, a simple subtraction operation is utilized, which takes the difference value between ceiling height and accessible height. In cases of the presence of ceiling hole, the adjacent regions that are sufficiently high are obtained through a breadth-first search strategy and the median height of them is utilized to update the elevation of the hole region. As illustrated in the height difference map in Fig. 6(a), the walls exhibit significant height variations and are marked in red, while the doorways have relatively smaller height difference, and near-zero height variations exist inside the room and are marked in blue.

Second, from the perspective of solid surface, the point density of walls is calculated by the rasterization way. Although the height difference map reveals the distribution of room boundaries, there remains scope for improvement in terms of detail. The decorative elements on the ceiling may leave unwanted traces in the height difference map. Additionally, deviations arising during data collection may lead to some supervoxels possessing a larger thickness, which is not conducive to locate the actual walls. To address these limitations, the point density map that can highlight the placement of walls [43] is introduced to cooperate with the height difference map. The process involves projecting all points within the non-horizontal supervoxels onto the ground and rasterizing them. The points within each grid cell are weighted on the basis of its height information and then accumulated to acquire the point density. Due to the likely presence of noise and obstacles near the floor, more weight is given to the points closer to the ceiling. After that, the height difference map I_h and the point density map I_d are subjected to a pixel-wise multiplication operation to fuse the occupancy information of the wall space and the density information of the wall surface. The result is mapped to eight-bit grayscale through min-max normalization, yielding the wall occupancy evidence map I_o in Fig. 6(c). The mathematical equations for this step are

as follows:

$$I_d = \sum_{p \in g(x,y)} \frac{h_c - h_p}{h_c - h_f} \quad (8)$$

$$R(x, y) = I_h \times I_d \quad (9)$$

$$I_o = \frac{R(x, y) - R_{\min}}{R_{\max} - R_{\min}} \times (L - 1) \quad (10)$$

where h_c , h_f , and h_p represent the height of the ceiling, floor, and an arbitrary point within a grid cell. The maximum and minimum values of multiplication result are denoted as $R_{\max}$ and $R_{\min}$, respectively. And L is the grayscale level, which equals to 2^8 for an eight-bit image.

By analyzing the arrangement patterns and attribute values among all pixels within a local neighborhood, it becomes feasible to infer the underlying wall orientation embedded in the occupancy evidence map. This feature plays a pivotal role in the selection of the anchor pixels and provides a directional constraint for subsequent boundary connection. Specifically, the orientation information of each pixel is obtained by the weighted principal component Analysis (WPCA) in 2D space. Pixels with prominent occupancy evidence are frequently situated at room boundaries, thereby contributing more to the determination of wall orientation. Based on the weight and position of eight adjacent pixels, WPCA constructs a covariance matrix that yields a set of principal directions after decomposition. In particular, the eigenvector associated with the largest eigenvalue represents the most significant extension direction of the current pixel, which is consistent with the orientation of the wall structure. According to the distribution pattern of neighboring pixels and the magnitude of the angle with the positive horizontal axis, the orientation values $\vec{v}_m$ are quantized into four uniform intervals using the formula below. Each of the available pixels is given one of the corresponding labels zone to generate the boundary orientation map in Fig. 7(a)

$$\text{zone} = \left\lceil \frac{\left(a \tan \left(\vec{v}_m \cdot y, \vec{v}_m \cdot x \right) + 9\pi/8 \right) \bmod \pi}{\pi/4} \right\rceil \quad (11)$$

where the symbol $\lceil \cdot \rceil$ denotes taking an integer upwards, and mod function represents the modulo operation. As illustrated in Fig. 7(b), each of the four intervals covers a quarter of a circle and is associated with the cardinal orientation of walls.

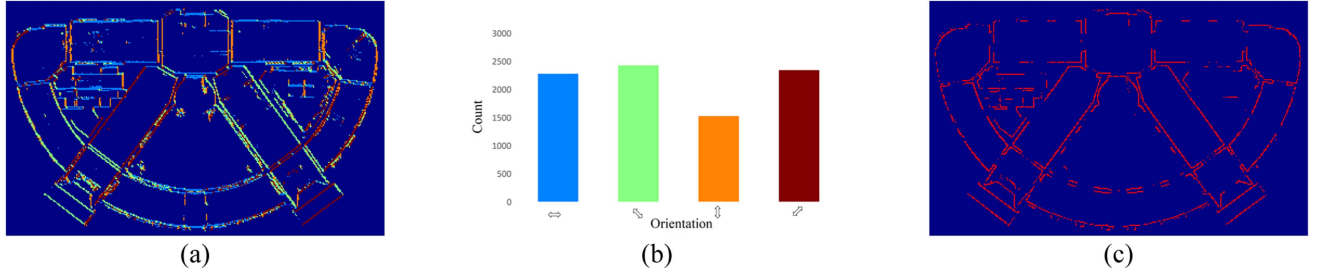


Fig. 7. Information extracted from wall occupancy evidence map: (a) boundary orientation map, (b) quantized result of orientation, and (c) anchor pixels.

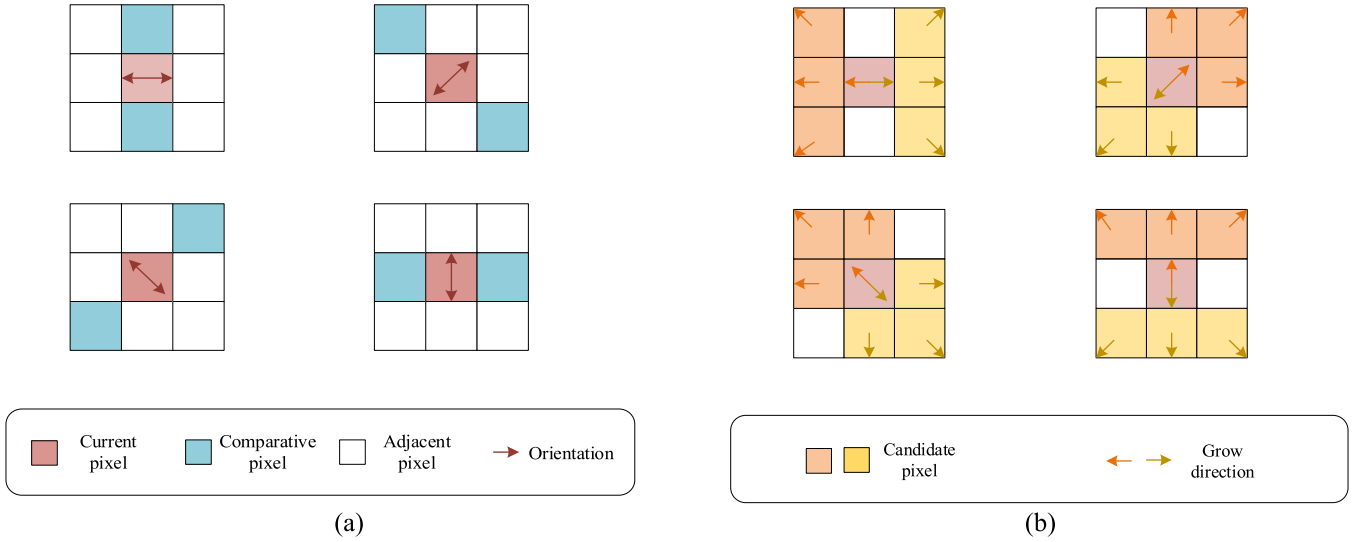


Fig. 8. Selection of pixels under orientation constraint: (a) extraction of anchor pixels, and (b) extension of pixels.

Subsequently, the anchor pixels are extracted from the wall occupancy evidence map with the assistance of boundary orientation and shown in Fig. 7(c). These pixels, which exhibit the maximal value within the local region, are most likely to indicate the exact location of walls and thus serve as crucial starting points for boundary linking. A non-maximum suppression strategy is applied to accurately pinpoint the anchor pixels. Specifically, for each unvisited pixel in the occupancy evidence map, a comparative analysis is conducted between it and two adjacent pixels that are perpendicular to its orientation. As the schematic illustrates in Fig. 8(a), in case of a horizontally oriented pixel, the two neighbors come from directly above and below it, respectively. If the value of current pixel is the biggest and exceeds a predefined threshold over one adjacent pixel, this pixel is deemed to be locally prominent and is consequently labeled as an anchor point. The median absolute deviation, which is robust to noise, is borrowed to automatically set the value of the threshold T_o and the calculation formula works as follows:

$$T_o = \text{med}(X) + \delta * \text{med}(|X_i - \text{med}(X)|) \quad (12)$$

where X denotes a set that contains the values of all nonzero pixels in the occupancy evidence map and X_i represents an individual sample within the set. The function med is used to

calculate the median value, while the constant factor δ controls the strictness of the threshold selection.

2) *Pixel Linking*: The next important task involves the design of a heuristic tracking strategy to extend the anchor pixels, aiming to obtain clean and continuous wall boundaries. This strategy is inspired by the previous research on line segment detection [44], [45], [46]. The pseudocode of the execution flow is listed in Algorithm 1. The local attribute information, such as point density or vertical height, is frequently utilized to rigorously infer whether a single grid corresponds to the wall [47]. But in complex environments, they are susceptible to noise and tend to produce fragmented wall boundaries. In reality, walls exhibit a high degree of continuity in 2D projected image, with gaps only at a few locations including doorways or occlusions. Therefore, the designed strategy starts with the prominent anchor pixels, and progressively extends outward under the guidance of the boundary orientation map and occupancy evidence map, thereby identifying high-quality boundaries of indoor rooms. Specifically, each connected component obtained from the initial room partitioning is treated as an operation unit for refinement process. The morphological dilation is performed on the selected unit to generate a mask, which confines the search scope for boundary pixels. The anchor pixels covered

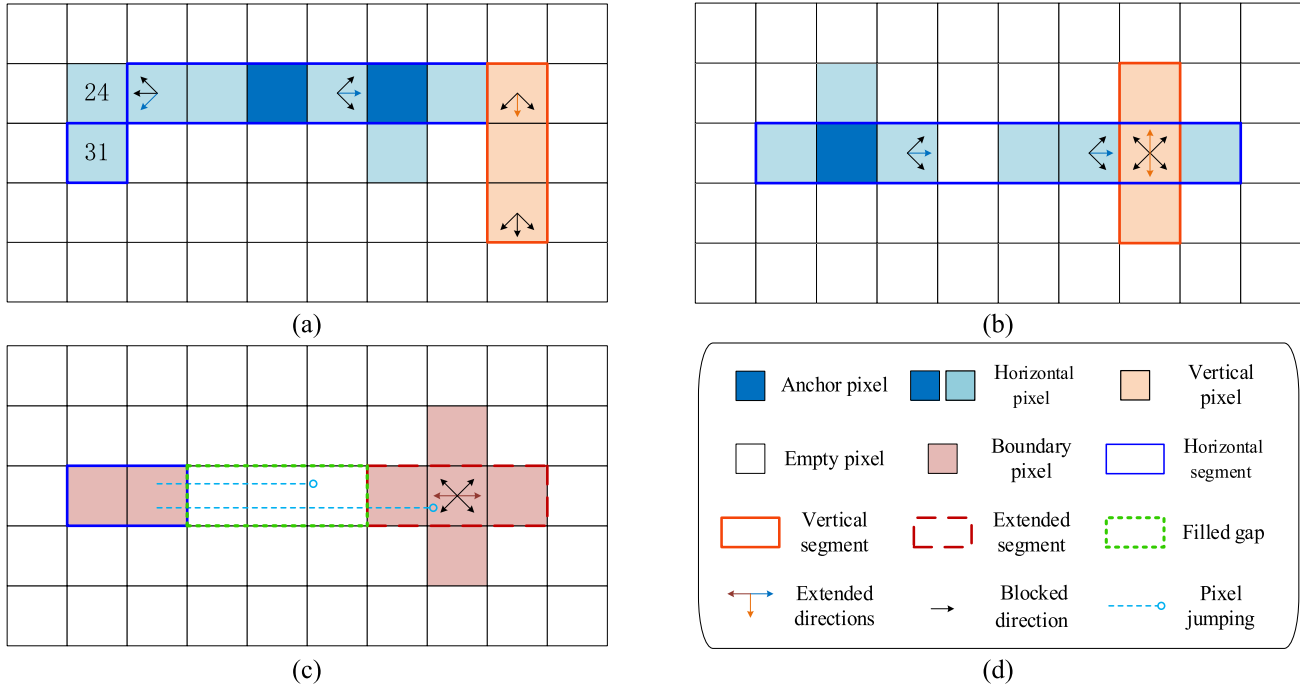


Fig. 9. Different ways of pixels linking: (a) sequential linking, (b) intermittent linking, (c) jump linking, and (d) annotation symbols.

by the mask are collected and sorted in descending order based on their values. Then initiating from the anchor point with the largest value, the exploration follows the corresponding wall orientation and moves toward both sides in an orderly manner. An intuitive illustration of the process is provided in Fig. 8(b), where in case of a horizontally oriented wall, the eligible pixels on the left are sequentially connected and followed by the right side. During each movement, three candidate neighbors in the current direction are evaluated. If an anchor pixel is encountered among these candidates, it is directly designated as the start point for the subsequent step. Otherwise, the neighbor with the highest value is selected to start the next move. This dynamic connection process persists until the orientation of the current pixel changes, the pixel value falls below a threshold, or the previously detected boundary is reached. In Fig. 9, blue solid-line boxes denote the detected horizontal wall segments, and orange solid-line boxes indicate the detected vertical wall segments.

After the movement is suspended, a collection of linearly arranged pixels is identified and constitutes a potential wall segment as depicted in Fig. 9(a). For a wall segment meeting the length requirement, the extension process restarts from the breakpoint and continues to move forward a certain distance along the direction of the wall. As exemplified in Fig. 9(b), once a prominent pixel that aligns with the travel direction is detected within the range, the new starting point shifts to this pixel and proceeds to sequentially connect adjacent pixels; if not, the movement in the current direction is completely terminated. In addition, upon encountering pivotal pixels with a change in direction while moving, these pixels are successively stored into a stack structure and serves as an initial point for subsequent tracking of walls with other orientations. Following

this, a series of potential wall segments triggered and connected by an individual anchor pixel are pieced together, and if the overall length is insufficient, pixels far from the outer boundary of the selected component are removed. This aims to filter out the trivial and isolated edges introduced by noise or furniture. After the anchor pixels in the mask image have been traversed, the focus shifts again to the wall segment with lengths exceeding a certain threshold.

The jump linking strategy is devised to lengthen the segments toward both ends to repair the gaps in the wall contour, the implementation of which is visualized in Fig. 9(c). Some exceptional scenarios may lead to notable gaps in the wall occupancy evidence map, such as doors near the ceiling or large obstructions. Thus, the jump linking strategy is necessitated to reinforce the integrity of wall boundaries. Specifically, a weighted least squares fitting method is adopted to precisely capture the dominant direction $(b, -a)$ of the wall segment. Under the constraint of this direction, the projection points of the end pixels (x_i, y_i) attempt to jump outwards with incremental steps and continuously track a sequence of pixels. If these pixels are sufficient in quantity and capable of forming a line parallel to the current wall segment, as shown by the red dotted box, the jumping process stops. Then the void pixels between the projected point (x_p, y_p) and the jumped point (x_n, y_n) , which are enclosed by green dotted box, are filled with the help of Bresenham's line algorithm to achieve seamless closure of gaps. The main mathematical formulas involved in this step are as follows:

$$\text{minimize}_{a,b,c} \sum w_i (ax_i + by_i + c)^2$$

Algorithm 1: Boundary pixels linking.

Input : I_o : Occupancy evidence map, I_b : Boundary orientation map,
 IR : Components after initial room partition

Output: I_w : Wall boundaries map

```

1   $Aps = \text{nonMaxSuppression}(I_o, I_b);$  // anchor pixels
2  for each component  $rc$  in  $IR$  do
3     $I_m, ob = \text{detMaskAndBoundary}(rc);$ 
4     $aps = \text{sortAnchorPixels}(I_m, Aps);$ 
5    for each anchor pixel  $ap$  in  $aps$  do
6       $stack = \text{initStack}(ap.coord, ap.val, ap.dir);$ 
7      while  $stack$  is not empty do
8         $curp = stack.pop();$ 
9        while  $curp.dir$  is not changed do
10          $nextp = \text{trackNextPixel}(curp, I_o, I_b);$ 
11         if  $nextp.dir = curp.dir$  AND  $nextp.val > threshold$  then
12            $se = \text{addToSegment}(nextp);$ 
13            $\text{labelPixel}(nextp, I_w);$  // Sequential linking
14         end
15       else
16          $stack.push(nextp);$ 
17         // Intermittent linking
18          $curp = \text{extendPixel}(nextp);$ 
19       end
20     end
21      $ws = \text{addToWallSegments}(se);$ 
22   end
23   // Pixels triggered by an anchor pixel
24    $wallchain = \text{addToPixelChain}(ws);$ 
25    $I_w = \text{removeNoisypixels}(wallchain, ob)$ 
26 end
27 for each wall segment  $se$  in  $ws$  do
28    $line = \text{weightFitLine}(se);$ 
29   for each jump step  $l$  in  $\Delta l$  do
30      $jumpixel = \text{jumpForward}(se.endpl, l, line, I_w);$ 
31     if  $\text{canGrow}(jumpixel, I_w)$  then
32        $holepixels = \text{Bresenham}(jumpixel, se.proj);$ 
33        $I_w = \text{fillHole}(holepixels, I_w);$  // Jump linking
34       break;
35     end
36   end
37 end

```

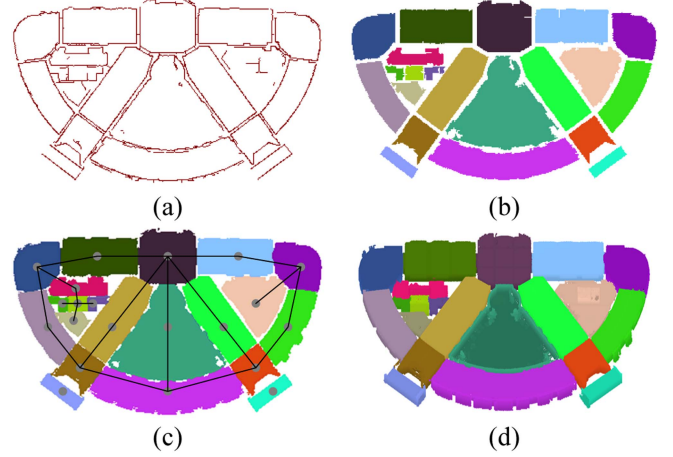


Fig. 10. Refined partitioning of rooms: (a) room boundary map, (b) room regions, (c) room segmentation map and connected network, and (d) final result of room partitioning.

with its internal pixels assigned a distinctive label. Thereafter, a wall-constrained region growing algorithm is employed to ensure that the unlabeled pixels are correctly categorized. During the growth process, labels of the pixels inside the room are propagated to adjacent unlabeled pixels, while the wall boundary pixels neither serve as starting points for propagation nor receive label from surrounding pixels. This is to avoid the incorrect situation where the room region crosses the actual walls. Following this, the remaining unlabeled pixels that mostly appear on the public walls or within minor regions are further processed using the topological information of the supervoxels. A room entity comprises multiple supervoxel projection areas, while a single projection area exhibits a certain probability of being distributed in several neighboring rooms. If the supervoxel projection area where a pixel to be classified belongs, or the adjacent projection areas are entirely located in the same room, the unclassified pixel directly inherits the label of this room. In other cases, a modified wavefront propagation method is employed to assign the most frequently occurring label in the vicinity, rather than the first accessed one to the unclassified pixel. After the label assignment of all pixels is completed, the room segmentation map is obtained in Fig. 10(c), which visually represents the layout and boundaries of indoor rooms. This is then combined with the information of narrow doorways and wall occupancy evidence map to obtain the connected network between rooms. The segmentation results in the horizontal plane are inversely mapped to the point cloud through coordinate correlations, thereby achieving the individualization of rooms in 3D space.

IV. EXPERIMENT

To validate the effectiveness of the proposed room partitioning method, this section selects some representative indoor point cloud datasets, which exhibit variability in terms of scale, layout, and complexity. Table I provides the key information of each dataset. Meanwhile, Fig. 11 offers a visual representation of the selected datasets. The first four datasets are derived from

$$s.t. \ a^2 + b^2 = 1 \quad (13)$$

$$\begin{pmatrix} x_p \\ y_p \end{pmatrix} = \begin{pmatrix} x_i \\ y_i \end{pmatrix} - \frac{(ax_i + by_i + c)}{a^2 + b^2} \cdot \begin{pmatrix} a \\ b \end{pmatrix} \quad (14)$$

$$\begin{pmatrix} x_n \\ y_n \end{pmatrix} = \begin{pmatrix} x_p \\ y_p \end{pmatrix} + \frac{\Delta l}{\sqrt{a^2 + b^2}} \cdot \begin{pmatrix} b \\ -a \end{pmatrix} \quad (15)$$

where w_i represents the weight and is equal to the pixel value of the wall occupancy evidence map. And Δl represents the step size of jumps, with its value gradually increasing over iterations.

3) *Label Propagation*: In the final step, all identified wall segments are superimposed to form a complete room boundary map, as illustrated in Fig. 10(a). Through the image differencing operation between this map and the outcome of initial room partitioning, the continuous pixels effectively cut off the tightly connected room sets. Then, multiple mutually independent regions are extracted from the differential image by using the connected component analysis. Any region satisfying the preset area and dimensional criteria is regarded as an individual room,

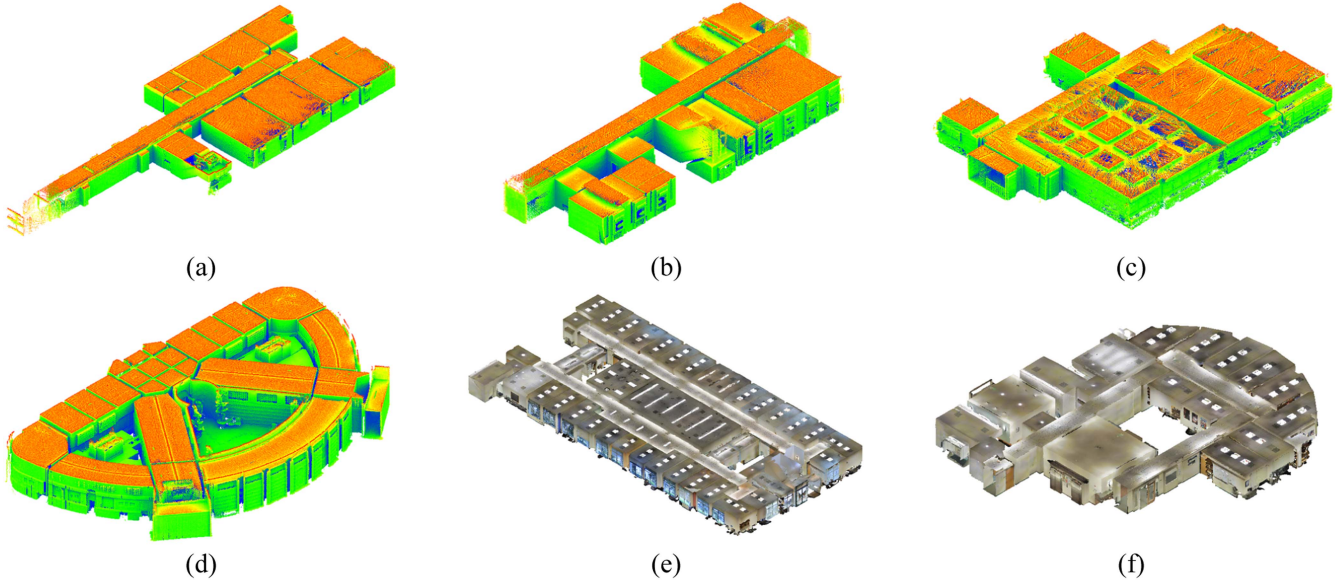


Fig. 11. Test dataset: (a) TUB2-1, (b) TUB2-2, (c) UoM, (d) Grainger Museum, (e) Area1, and (f) Area3.

TABLE I
KEY INFORMATION OF THE TEST DATASETS

Test dataset	Capture device	Points (million)	Area (m ²)	Height (m)	Clutter level
TUB2-1	Laser scanner	13.1	641.1	2.4	Low
TUB2-2	Laser scanner	8.5	592.3	6.0	Low
UoM	Laser scanner	12.6	479.8	3.8	Medium
Grainger Museum	Laser scanner	24.2	700.3	4.9	High
Area1	RGBD camera	44.0	1189.9	5.3	High
Area3	RGBD camera	18.7	752.5	3.1	High

TABLE II
CONFIGURATION OF PARAMETERS

Symbol	Description	Value
cs	Cube size	0.05 m
$knei$	Nearest neighbors	21
$\omega_s, \omega_n, \omega_c$	Weight coefficients	0.6, 1.0, 0.4
SR	Supervoxel resolution	1.0 m
gs	Grid size	0.1 m
$empty_{min}$	Min height of free space	0.5 or 0.8 m
p_{min}	Min pixel value for linking	1 or 3
len_{min}	Min length of wall segment	1.0 m
Δl	Jump step size	0.5, 0.7, 0.9, 1.1 m
ra_{min}	Min area for room	1.0 m ²
rh_{min}	Min height for room	1.8 m
rlw_{min}	Min length and width	0.8 m

the ISPRS Benchmark on Indoor Modeling [48], where the first and second datasets pertain to the upper and lower stories of building TUB2, respectively, and are connected by a staircase. UoM and Grainger Museum datasets are collected using mobile laser scanner in cluttered environments and include considerable noise. The latter two datasets, Area1 and Area3, are obtained from the Stanford 3D Large-Scale Indoor Scene Dataset [34] and captured by RGBD camera. Notably, in order to facilitate room partitioning, some datasets undergo preprocessing steps including coordinate reorientation and outdoor point removal, which can be realized via the previous methods [49].

The implementation and testing of the method are conducted on a laptop equipped with a 10th generation Intel Core i7 processor. The programming language is C++, and open-source PCL and OpenCV library tools are introduced to support the efficient processing of 3D point clouds and 2D images, respectively.

A. Parameter Setting

The configuration of parameters for different datasets is summarized in Table II. The vast majority of parameters exhibit stability across various datasets. Initially, the cube with a size of 0.05 m is used to downsample the input point cloud. The number of nearest neighbors is chosen to be 21 for the estimation of local normal vector and surface curvature. The resolution of supervoxel is delineated as 1.0 m, and the assignment of three weighting coefficients are 0.6, 1.0, and 0.4, respectively. The supervoxels are rasterized at a resolution of 0.1 m, which is consistent with the scale of subsequent 3D voxelization process. At the refined stage, the step size of jumping connection is progressively incremented from 0.5 to 1.1 m, allowing to cope with intricate and variable environments. A minimum area of 1.0 square meters is imposed for a room to be deemed acceptable, alongside the requirement that both the length and width exceed 0.8 m. In addition to constraining planar dimensions, a vertical

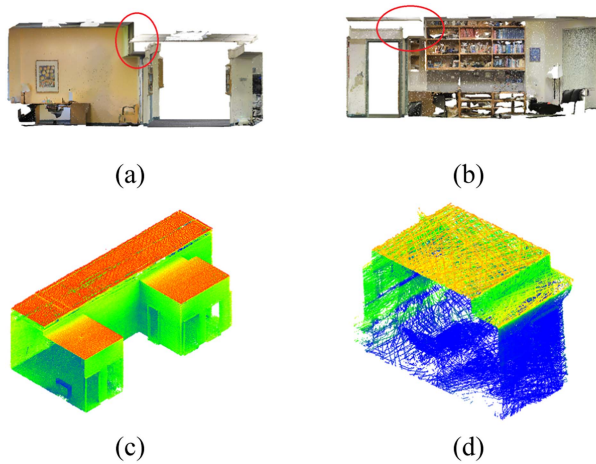


Fig. 12. Complex scenes in some datasets: (a) and (b) scattered points above the ceiling in Area1 and Area3, and (c) and (d) abrupt drop of ceiling elevation in TUB2-2 and UoM.

height requirement of 1.8 m for rooms is added. Meanwhile, given that a room is enclosed by the ceiling, floor, and walls, candidate rooms with severe missing vertical point clouds are discarded.

Besides, there are still two important parameters that need to be adjusted in accordance with the data characteristics. The first one is the height of free space, which directly affects the pixel value of the wall occupancy evidence map. As exemplified in Fig. 12, the aperture structures of the corridor result in some points dispersed above the ceilings of adjacent rooms. To prevent ignoring the wall space below, the height threshold is raised from the conventional 0.5 to 0.8 m in Area1 and Area3 datasets. The second one is the minimum pixel value during dynamic linking, which controls the sensitivity of wall boundary recognition. For TUB2-2 and UoM datasets, this parameter is incremented to 3 from the default value of 1, to mitigate errors arising in places where the ceiling abruptly drops in the same room.

B. Evaluation Results

The performance of the proposed method in various indoor environments is illustrated in Fig. 13. Each dataset is presented with three subfigures, the first column is the reference benchmark, which are either cropped after careful observation or obtained from existing ground truth. The second column shows the core regions of the extracted rooms, with the black lines delineating the connected wall boundaries. The third column focuses on the partitioned room point cloud, wherein each room is given a unique label and distinguished by random color. The experimental outcomes indicate that the proposed method achieves remarkable performance, as almost all test datasets can be accurately identified and partitioned, even in the presence of complex structures such as curved walls that deviate from the Manhattan assumption. From the perspective of free space and solid surface, the proposed method integrates multiple sources of information, such as spatial voxel distribution, point density, and point height to construct the wall occupancy evidence map,

which greatly reduces the interference of unstructured objects. Specifically, the first story of TUB2 building is partitioned into 15 distinct point cloud subsets, whereas the second story undergoes a division into 9 rooms. The incomplete thin wall in the middle of the long corridor is successfully detected, thus avoiding the incorrect fusion of the rooms on both sides. In the dataset of TUB2, the two spaces closely connected to the left corridor are not separated due to the lack of distinct wall evidence. In terms of the functional utility and visual inspection, it is indeed more rational to consider these two spaces as part of the corridor, rather than independent entities. Seven rooms are detected from the UoM dataset, including a winding public corridor. Despite the generated wall occupancy evidence map has defects due to the interference from moving pedestrians, ventilation facilities, and bookshelves in rooms. The method is still able to discover continuous and clean wall boundary pixels under orientation constraint. Notably, the Grainger Museum dataset has an abundance of curved walls and cluttered booths, resulting in a high degree of complexity to the internal environment. This poses a significant challenge for room segmentation methods that rely on wall plane fitting. Conversely, the proposed method effectively separates the tightly connected rooms without direct extraction of planes and yields 21 distinct units. The curved walls with constantly changing orientations can be captured in the process of pixel linking, thereby preventing the formation of fragmented segments and generating continuous boundaries. The datasets of Area1 and Area3 further validate the capability of the method in handling large-scale indoor scenes. There are 45 and 23 rooms extracted from these two datasets, with only one more room found in Area1 in comparison with the reference data. The over-segmentation problem occurs in the stairwell and is marked by a blue arrow in Fig. 13(e). The height of this staircase is much greater than that of other rooms and the integrity of point cloud is poor. The T-shaped wall at the center makes a clear imprint in the occupancy evidence map and covers the opening directly below, thus causing the erroneous division of the stairwell into two rooms. In addition, a flaw observed in Area3 is the classification deviation of some points in Fig. 13(f), as highlighted by a red arrow. This is due to the severe obstruction of the wall behind the door, coupled with the interference from the nearby cabinets, which leads to the connected boundary diverging from the shared wall. Consequently, some pixels at the junction of rooms are misclassified. Although the subsequent label propagation process can improve this situation to a certain extent, it is still not possible to ensure that all of the boundary points are accurately categorized into the corresponding rooms. Despite encountering few defects in several datasets, the proposed method maintains a high level of accuracy, robustness, and versatility across a wide range of indoor environments.

In addition to intuitive visual evaluation, this section also quantitatively analyzes the performance of the method on six experimental datasets. Table III presents a variety of quantitative metrics, encompassing the accuracy of the partitioned point cloud, the deviation in room counts, and the execution time of the method. Among these, accuracy stands as a pivotal global metric, aiming to measure the proportion of correctly partitioned samples relative to the total samples at the point level. To achieve

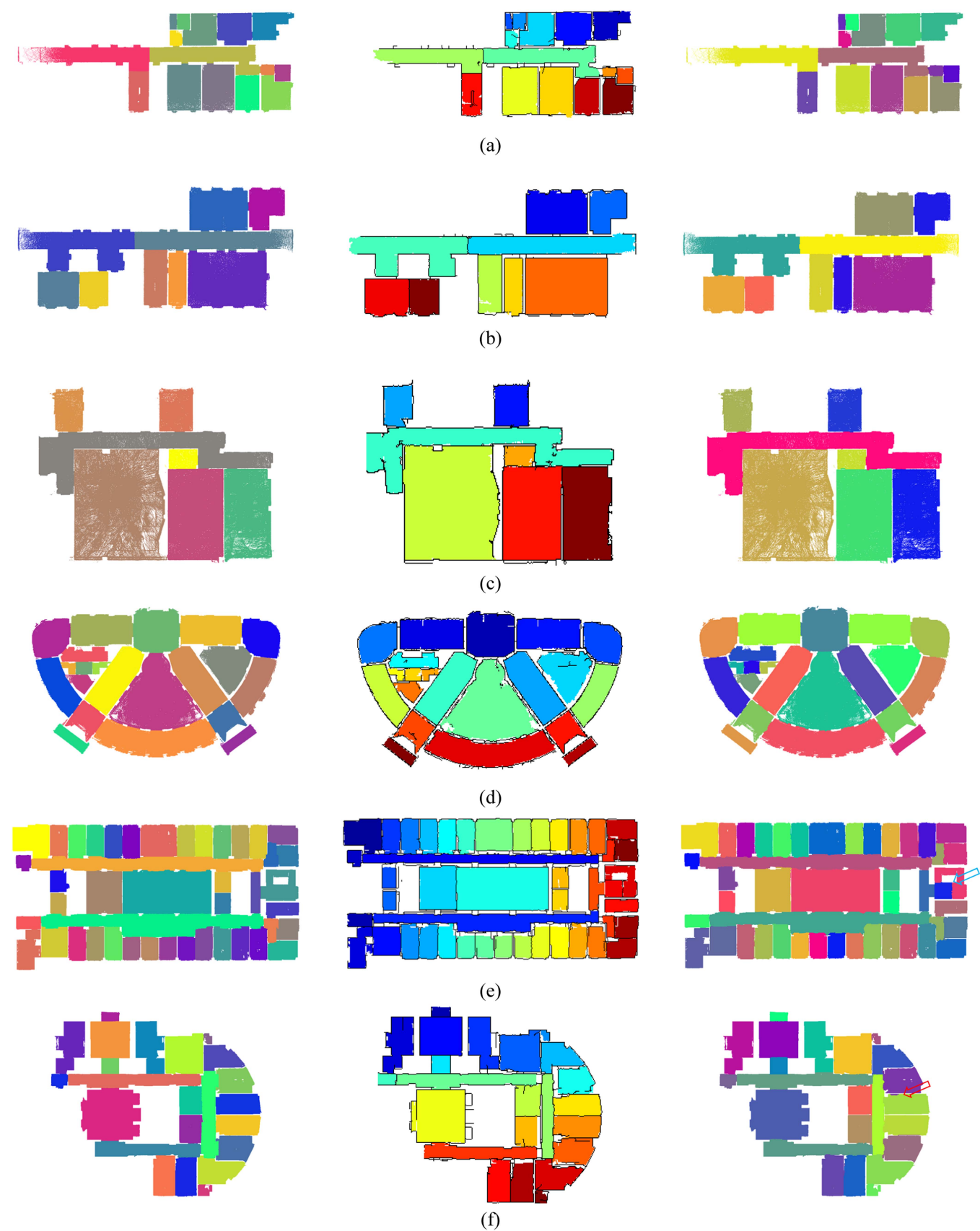


Fig. 13. Partitioning results of test datasets. (a) TUB2-1, (b) TUB2-2, (c) UoM, (d) Grainger Museum, (e) Area1, and (f) Area3.

TABLE III
QUANTITATIVE EVALUATION RESULTS

Dataset	A_{PPC} (%)	N_{actual}	$N_{extracted}$	D_{RC}	T_E (s)
TUB2-1	98.86	15	15	0	7.2
TUB2-2	99.12	9	9	0	8.6
UoM	99.17	7	7	0	8.5
Grainger Museum	99.02	21	21	0	18.8
Area1	93.65	44	45	1	19.3
Area3	96.31	23	23	0	7.5

this evaluation, a bidirectional one-to-one mapping of rooms is deduced on the basis of the criterion of maximum overlap, ensuring that each extracted room uniquely corresponds to an actual room and each actual room is associated with at most one extracted room. The intersecting point clouds within the pairwise matched rooms are regarded as correctly partitioned samples. The total samples originate from all points in the initial point cloud, including not only correctly partitioned points, but also those erroneously assigned to other rooms as well as unclassified points. The deviation in room count is a metric used to quantify the discrepancy between the number of rooms extracted by the method and the actual number of rooms, with both oversegmentation and undersegmentation contributing to an increased deviation value. The execution time directly reflects the efficiency of the method in practical applications. The accuracy of the partitioned point cloud A_{PPC} and the deviation in the number of rooms D_{RC} can be formally expressed as

$$A_{PPC} = \frac{\sum_{i=1}^{N_{matched}} (C_i^{extracted} \cap C_i^{actual})}{\sum_{j=1}^{N_{actual}} C_j^{actual}} \quad (16)$$

$$D_{RC} = |N_{extracted} - N_{actual}| \quad (17)$$

where $N_{matched}$, N_{actual} , and $N_{extracted}$ denote the number of pairwise matched rooms, actual rooms in reference data, and extracted rooms, respectively. Similarly, $C_i^{extracted}$ and C_i^{actual} represent the point cloud in one of the extracted rooms and actual rooms, respectively.

The quantitative outcomes also significantly demonstrate that the proposed method has excellent performance. It achieves an accuracy of about 99% as well as no quantity deviation in most experimental data, even for the Grainger Museum dataset with complex structures. However, a notable decline in accuracy to 93.65%, along with an increase in deviation to one is observed in the dataset of Area1. This is primarily attributed to the stair region being excessively partitioned into two separate rooms. Area3 dataset has zero deviation in room count but shows a relatively low accuracy of 96.31%. This is due to the fact that some points of the long corridor are distributed above the ceiling of the adjacent rooms, and form overlapping raster after downward projection. All points in the same raster are treated as a single entity when the segmented results are back propagated to the original point cloud, which results in the misclassification of some points. This issue also exerts a negative influence on the accuracy of Area1. Besides, unexpected factors such as solid door occlusion, furniture interference, and wall

holes have an effect on the identification of room boundary in some indoor scenes like TUB2-1 and Area3, which triggers a small number of points to be incorrectly categorized. However, these factors exhibit limited influence on the overall accuracy. Furthermore, the analysis reveals that laser datasets achieve slightly higher average accuracy at 99.04%. And the method maintains strong performance on RGBD data, with an average accuracy of 94.98%, proving its resilience to sensor types.

The execution time T_E of the method is closely related to the characteristics of the experimental data, particularly the number of points, the complexity of the environment, and the spatial extent of the building. The average time for room partitioning in six different datasets is kept within 12 seconds. Among them, the highly complex Grainger Museum and Area1 datasets, which contain millions of points and cover hundreds of square meters, consume the longest time of about 19 seconds. For the remaining datasets, the method takes less than 10 seconds to complete the partitioning. It is worth noting that supervoxel segmentation occupies more than 80% of the overall runtime, significantly exceeding the time consumed in the steps of initial and refined room partitioning. This is due to the need to calculate the local features of each point and the similarity between neighboring points in the generation of supervoxels. Overall, the proposed method is capable of partitioning the point cloud into rooms within a shorter period of time.

C. Comparison Results

To further reveal the advantages of the proposed method, this section conducts a comparative analysis with mainstream methods on a series of challenging datasets. The room partitioning results for the TUB2-1 dataset are shown in Fig. 14, where the first subplot contains the scanning trajectory and the point cloud with the ceiling removed to clearly present the details of the indoor room layouts. Although this building adheres to the typical Manhattan world assumption, extra complexities are introduced to the partitioning task due to thin walls and open stairwells. Morphology-based method [23] separates connected regions through iterative erosion operations but struggles to distinguish between tightly coupled corridors and rooms, resulting in under-segmentation problem as evidenced in Fig. 14(b). Wu et al. [29] devise a method combining cloth simulation filtering and regular grid analysis to accommodate buildings with varying room heights or slanted ceilings. However, the method easily overlooks the point cloud in the transitional passages and also suffers from the under-segmentation at the long corridor. Yang et al. [40] proposed a method on the basis of sparse voxels and sphere packing to perform room partitioning directly in 3D space, bypassing the projection of the point cloud. Nevertheless, this method is susceptible to disturbances from obstacles like pedestrians and steps in the environment during distance transformation, leading to the problem of oversegmentation in the corridor and stairwell, as demonstrated in Fig. 14(d). Tang et al. [31] employ an improved morphological method to spatially decompose the binary image generated by the superposition of original point clouds and vertical structural primitives. Although this method is somewhat robust to noise, the decomposition

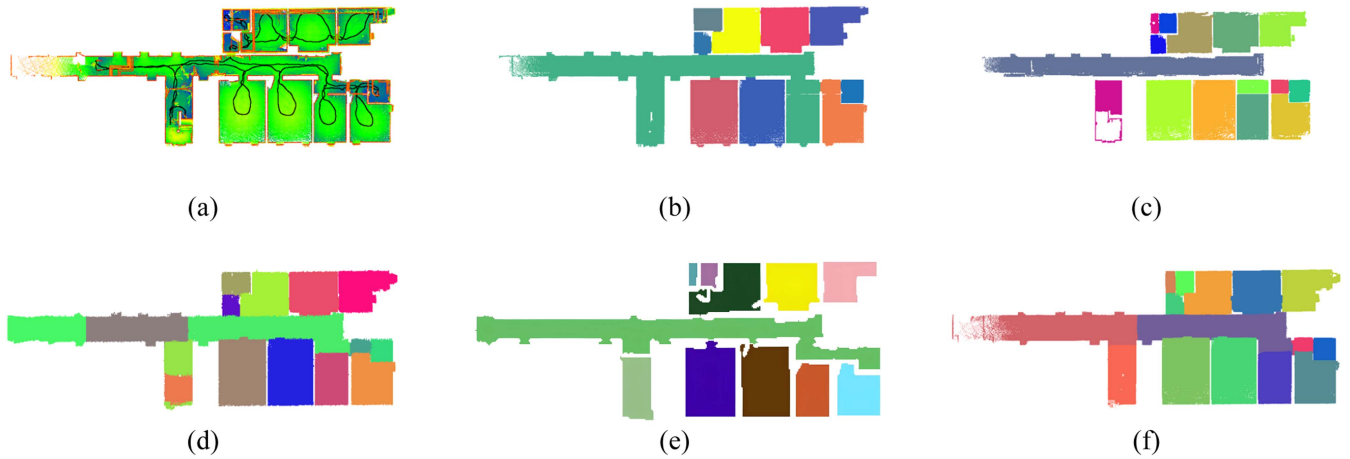


Fig. 14. Partitioning results of different methods on TUB2-1 dataset: (a) test dataset, (b) morphology method, (c) Wu et al.'s method, (d) Yang et al.'s method, (e) Tang et al.'s method, and (f) proposed method.

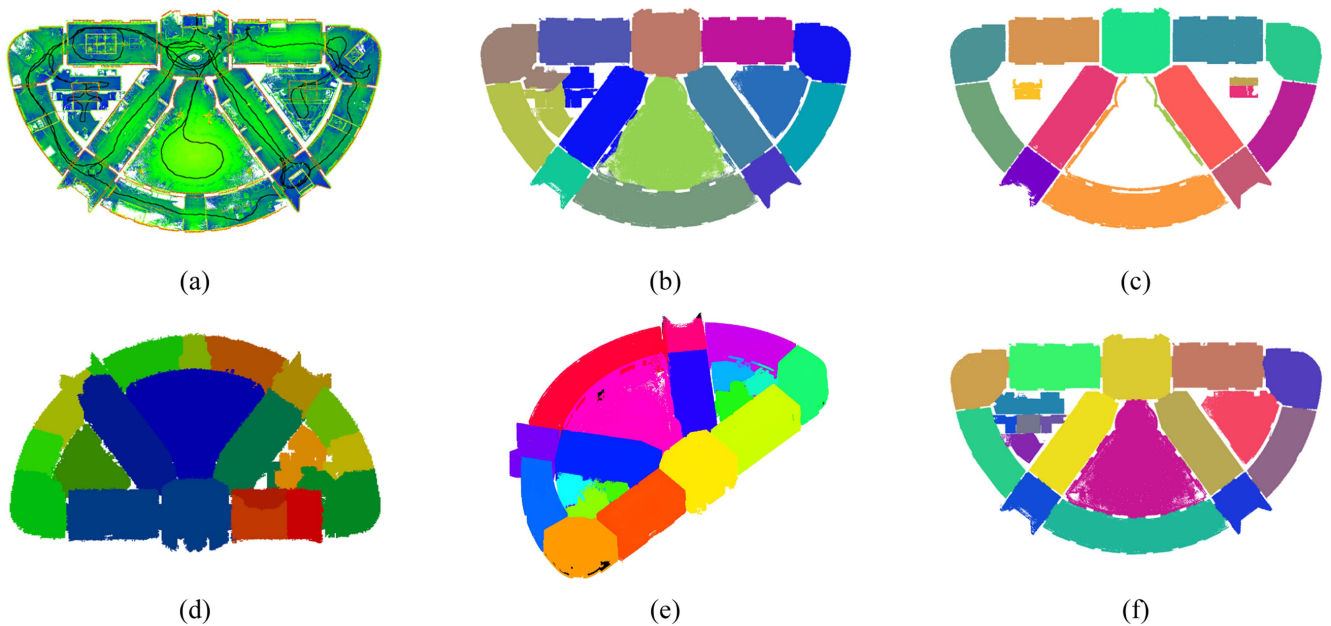


Fig. 15. Partitioning results of different methods on Grainger Museum dataset: (a) test dataset, (b) distance transform method, (c) Wu et al.'s method, (d) Yang et al.'s method, (e) Martens et al.'s method, and (f) proposed method.

result depends on the planar primitives detected from the unstructured point cloud, and when the wall points are missing severely, it may yield the undersegmented corridor in Fig. 14(e). In contrast, the proposed method outperforms other methods on the current dataset by linking the anchor pixels to obtain continuous wall boundaries and eliminating the need to detect planes.

The room partitioning results obtained by diverse methods on the Grainger Museum dataset are visually presented in Fig. 15. This building is particularly challenging as the spatial layout does not follow the Manhattan structural assumption and the indoor environment is chaotic. The method based on distance transform [23] treats regions far from image edges as potential room centers. Through multiple adjustments of projection

resolution and area threshold, most rooms are separated but some of the boundaries are quite different from reality, especially the small rooms on the left side of Fig. 15(b). Unlike directly selecting the point cloud within a fixed offset space near the ceiling, Wu et al. [29] utilize the cloth simulation filtering algorithm to better handle buildings with inconsistent room heights. However, the extracted ceilings may not be complete when the scene exhibits significant variation in elevation, which leads to a considerable lack of point cloud in the result of Fig. 15(c). The method of Yang et al. [40] has certain advantages in tackling complex buildings that contain nested rooms and cross-floor spaces. Nevertheless, its drawback lies in the dependency on trajectory data and sensitivity to interference factors like occlusions, noise, and reflections, which may greatly

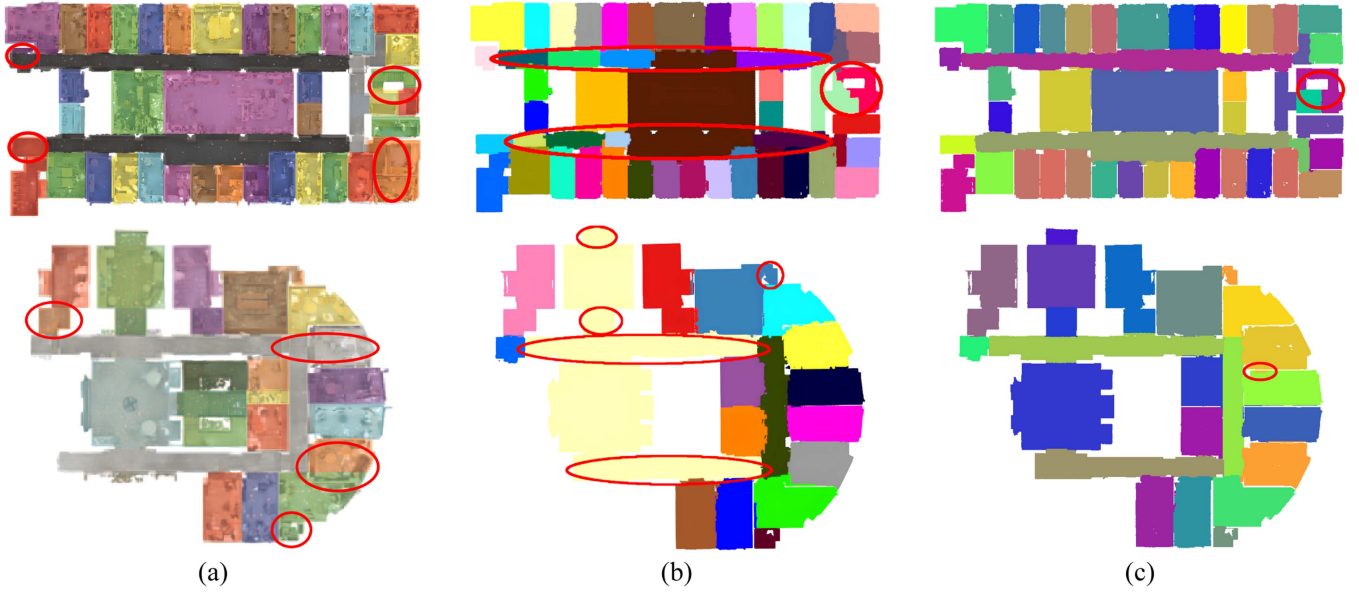


Fig. 16. Partitioning results of different methods on Area1 and Area3 dataset: (a) Armeni et al.'s method, (b) Bobkov et al.'s method, and (c) proposed method.

impact the overall partitioning quality, as shown in Fig. 15(d). The strategy of Martens and Blankenbach [32] is to create non-overlapping rooms by gradually growing outward from the seed regions with the assistance of the wall grids. The method achieves robust and accurate results on some buildings with atypical Manhattan assumption. However, it fails to avoid the influence of vertical furniture and narrow spaces and causes the incorrect division of the small rooms in Grainger Museum dataset, as demonstrated in Fig. 15(e). The proposed method eliminates the necessity for extra scanning trajectory, relying solely on spatial distribution information of the point cloud to reveal indoor room layout. Experimental results of this challenging dataset also indicate that the proposed method surpasses alternative approaches in performance.

In addressing the large-scale and multi-room datasets Area1 and Area3, the comparative results of different methods are presented in Fig. 16. The red circles in the figures highlight where the errors occurred, which include the instances of oversegmentation, undersegmentation, and inaccurate boundaries. The left subfigure originates from the method proposed by Armeni et al. [34], which leverages density histograms to search for voids enclosed by wall entities as a way to parse the building into multiple independent rooms. This method exhibits robustness in cluttered environments since there is no requirement for detecting the surface information from input point cloud. However, it is not suitable for rooms that deviate from the main walls of the building or have more freeform shapes. Bobkov et al. [35] present a room partitioning method grounded in anisotropic potential fields. It does not depend on the Manhattan world assumption or scanning location information, making it applicable to a wider range of irregular indoor environments. Nevertheless, the computational costs associated with estimating potential field values and unsupervised clustering are relatively high. And as shown in Fig. 16(b), the division

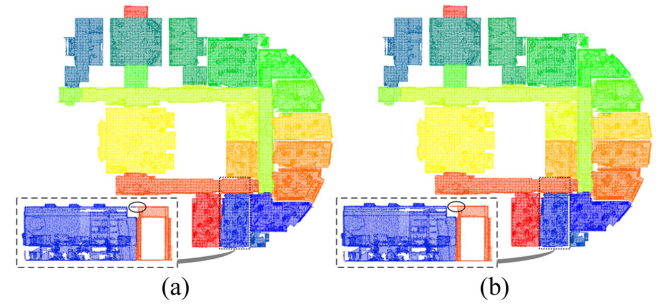


Fig. 17. Improved room partitioning result on Area3 dataset: (a) original result, and (b) improved result.

performance of this method in the corridor is poor. By comparison, the proposed method is not limited to Manhattan buildings and significantly mitigates the partitioning flaws on the test datasets.

V. DISCUSSION

Comparison analysis demonstrates that the proposed method outperforms other methods in real-world scenarios with curved structures, elongated corridors, and thin walls, showing strong applicability and robustness. As shown in the lower right corner of Fig. 1, the partitioning results not only facilitate the analysis and reconstruction of individual entities with reduced complexity and smaller data volumes, but also assist in extracting connectivity relationship between distant rooms for indoor navigation. Although it still has limitations to deal with some special indoor scenes. Especially when neighboring rooms overlap together after projection, part of the point cloud will be misclassified because the points within the same raster grid are considered as a whole. As illustrated in the box of Fig. 17, the hole structure at the top of the corridor causes some points to be

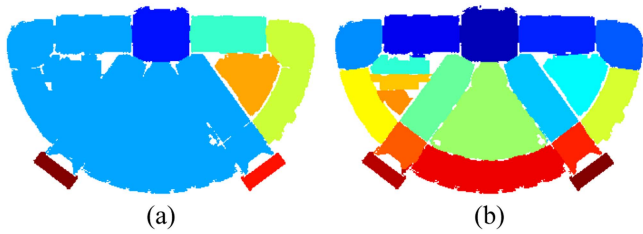


Fig. 18. Ablation experiment: (a) only initial partitioning, and (b) only refined partitioning.

scattered on the ceiling of adjacent rooms. This problem may be solved by introducing the clustering strategy in 3D space to reclassify the overlapping points. Specifically, the corridors that connect multiple regions in the room segmentation map are first identified. Then, focus is shifted to the point cloud within a certain distance range from the ceiling to minimize interference and reduce computational load. Next, the point cloud at the top of the corridor is used as the clustering centroid and gradually incorporates the nearby points from the surrounding rooms. Finally, all points within a single cluster are assigned the corresponding room label, and the improved result is shown in Fig. 17(b). In terms of quantitative evaluation metrics, the accuracy of the Area1 and Area3 datasets has been improved by 0.37% and 0.48%, respectively. This demonstrates that the introduced 3D spatial clustering strategy effectively addresses the issue of room overlap and enhances the robustness of the method.

The proposed room partitioning method consists of two main phases. To validate the importance and necessity of each phase, ablation experiments are conducted by independently evaluating their impact on overall performance. The initial partitioning phase highlights gaps between rooms. However, due to noise in point cloud and complexity of indoor environment, the majority of rooms remain tightly connected as shown in Fig. 18(a). While the refined partitioning phase can identify most rooms through anchor pixel linking, it fails to effectively segment small rooms in cluttered areas in Fig. 18(b), affecting the accuracy of the results. By integrating both phases, more accurate and reliable results are achieved in Fig. 15(f), in which room count deviation is zero. The initial phase removes narrow structures and doorways, providing a cleaner input for the refined stage, which then optimizes the wall boundaries to produce high-quality partitioning.

Furthermore, we analyze the impact of key parameters such as supervoxel resolution and jump step size. The partitioning results in Fig. 13 indicate that a supervoxel resolution of 1.0 m and a jump step size incrementally adjusted from 0.5 to 1.1 m offer an optimal balance between detail preservation and computational efficiency, and are applicable to a wide range of test datasets. The selection of supervoxel resolution is primarily based on the minimum area of rooms, while the determination of the jump step is informed by the common width of doors.

Supervoxels are spatially contiguous clusters of points with homogeneous geometric features and serve as the main operating objects in the initial partitioning phase. The resolution

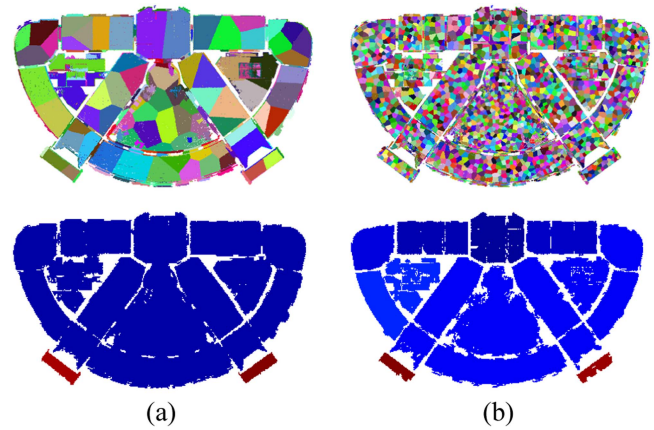


Fig. 19. Effect of supervoxel resolution: (a) supervoxel resolution at 2.0 m, and (b) supervoxel resolution at 0.5 m.

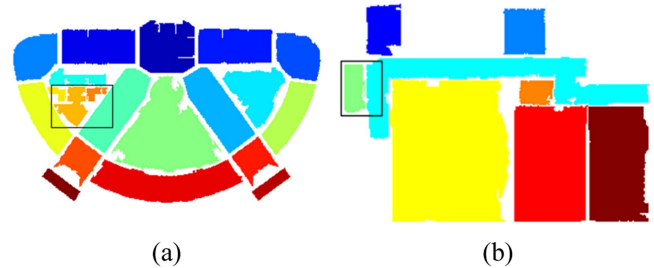


Fig. 20. Effect of jump step: (a) jump step at 0.3 m, and (b) jump step at 1.3 m.

determines the scale of supervoxels in space, which directly affects the removal of narrow structures and door passages. A range of resolution values, from relatively coarse to fine are tested for comparison. The top and bottom rows of Fig. 19 show the supervoxel segmentation results and the initial room partitioning results, respectively. As shown in Fig. 19(a), a low supervoxel resolution of 2.0 m causes supervoxels to span across adjacent rooms. While this increases processing speed, it leads to the loss of important boundary information, and as a result, some door passages and narrow structures fail to be detected. A high supervoxel resolution of 0.5 m, as depicted in Fig. 19(b), retains more details but tends to oversegment. This can result in the erroneous removal of areas that are not actually door passages or narrow structures, such as ceiling indentations, potentially splitting a single room into multiple parts.

The jump size is a critical parameter in the refined partitioning phase. Typically, doorways connecting different spaces leave distinct marks on the supervoxel projection map or the wall occupancy evidence map. These features serve as important cues to close openings. However, special cases, such as doors positioned close to the ceiling or significant point cloud loss, can create gaps that are challenging to identify. In such scenarios, the use of jump linking becomes essential to improve the completeness of wall boundaries and the jump step of pixels directly determines whether the gaps can be correctly filled. Fig. 20 shows the core region of rooms obtained with different jump steps. As shown

in Fig. 20(a), if the jump step is too short, the gap may fail to be detected. The doors between the two neighboring small rooms are not closed, causing them to be closely connected. As shown in Fig. 20(b), when the jump step is excessively long, the gap may be incorrectly filled. The wall boundary line is extended into the interior of the corridor, resulting in the long corridor being over divided. To effectively adapt to the complex and variable indoor environments, a strategy of gradually increasing jump step is adopted. Starting from the smallest step and gradually increasing it, each distance is tested until a qualified gap is identified or the maximum step is reached.

VI. CONCLUSION

This article proposes an automated room partitioning method for indoor point cloud, which consists of two main stages. In the initial partitioning phase, narrow structures and doorways are eliminated by morphological erosion and geometric feature analysis to highlight the gaps between rooms. Subsequently, in the refinement stage, continuous and clean room boundaries are generated by connecting the anchor pixels in the wall occupancy evidence map. Finally, the individualization of rooms is achieved by propagating the segmentation results on the image to the overall point cloud through inverse mapping. Six datasets with varying structural layout, scale, and complexity are selected to validate the performance of the proposed method. Through qualitative, quantitative, and comparative analyses, the results show that the proposed method accurately identifies almost all rooms in each dataset while effectively reducing the negative effects of noise, occlusion, and data loss. The average accuracy of partitioned point cloud is close to 98%, and the longest execution time is controlled within 20 seconds. Besides, it overcomes the limitations of the Manhattan-world assumption and eliminates the need for additional data. The deep-learning methods directly process unordered point cloud and achieve accurate semantic segmentation in cluttered environments. The extracted indoor structural elements are helpful for room partitioning. But there are some challenges with these methods, such as training requirements for large amounts of data and limited generalizability to different indoor layouts. Future work will include the integration of an efficient and lightweight network.

REFERENCES

- [1] R. Romero-Jarén and J. J. Arranz, "Automatic segmentation and classification of BIM elements from point clouds," *Automat. Construction*, vol. 124, Apr. 2021, Art. no. 103576, doi: [10.1016/j.autcon.2021.103576](https://doi.org/10.1016/j.autcon.2021.103576).
- [2] Z. Kang, J. Yang, Z. Yang, and S. Cheng, "A review of techniques for 3D reconstruction of indoor environments," *ISPRS Int. J. Geo-Inf.*, vol. 9, no. 5, 2020, Art. no. 330, doi: [10.3390/ijgi9050330](https://doi.org/10.3390/ijgi9050330).
- [3] G. Pintore, C. Mura, F. Ganovelli, L. Fuentes-Perez, R. Pajarola, and E. Gobbetti, "Automatic 3D reconstruction of structured indoor environments," in *Proc. ACM SIGGRAPH 2020 Courses*, 2020, pp. 1–218, doi: [10.1145/3388769.3407469](https://doi.org/10.1145/3388769.3407469).
- [4] S. Ikehata, H. Yang, and Y. Furukawa, "Structured indoor modeling," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 1323–1331, doi: [10.1109/ICCV.2015.156](https://doi.org/10.1109/ICCV.2015.156).
- [5] H. Tran, K. Khoshelham, A. Kealy, and L. Díaz-Vilariño, "Shape grammar approach to 3D modeling of indoor environments using point clouds," *J. Comput. Civil Eng.*, vol. 33, no. 1, 2019, Art. no. 04018055, doi: [10.1061/\(asce\)cp.1943-5487.0000800](https://doi.org/10.1061/(asce)cp.1943-5487.0000800).
- [6] A. Adán, B. Quintana, S. A. Prieto, and F. Bosché, "An autonomous robotic platform for automatic extraction of detailed semantic models of buildings," *Automat. Construction*, vol. 109, 2020, Art. no. 102963, doi: [10.1016/j.autcon.2019.102963](https://doi.org/10.1016/j.autcon.2019.102963).
- [7] Y. Dai, J. Gong, Y. Li, and Q. Feng, "Building segmentation and outline extraction from UAV image-derived point clouds by a line growing algorithm," *Int. J. Digit. Earth*, vol. 10, no. 11, pp. 1077–1097, Nov. 2017, doi: [10.1080/17538947.2016.1269841](https://doi.org/10.1080/17538947.2016.1269841).
- [8] Y. Verdier, F. Lafarge, and P. Alliez, "LOD generation for urban scenes," *ACM Trans. Graph.*, vol. 34, no. 3, pp. 1–14, 2015, doi: [10.1145/2732527](https://doi.org/10.1145/2732527).
- [9] K. Pantazatou, J. Kanteris, P.-O. Olsson, J. L. Nyborg, and L. Harrie, "Input data requirements for daylight simulations in urban densifications," *Urban Inform.*, vol. 2, no. 1, pp. 1–27, Jun. 2023, doi: [10.1007/s44212-023-00024-6](https://doi.org/10.1007/s44212-023-00024-6).
- [10] M. Mielles, M. Magnusson, and A. J. Lilienthal, "A method to segment maps from different modalities using free space layout MAORIS: Map of ripples segmentation," in *Proc. IEEE Int. Conf. Robot. Automat.*, Sep. 2018, pp. 4993–4999, doi: [10.1109/ICRA.2018.8461128](https://doi.org/10.1109/ICRA.2018.8461128).
- [11] M. Luperto, T. P. Kucner, A. Tassi, M. Magnusson, and F. Amigoni, "Robust structure identification and room segmentation of cluttered indoor environments from occupancy grid maps," *IEEE Robot. Automat. Lett.*, vol. 7, no. 3, pp. 7974–7981, Jul. 2022, doi: [10.1109/LRA.2022.3186495](https://doi.org/10.1109/LRA.2022.3186495).
- [12] L. Fermin-Leon, J. Neira, and J. A. Castellanos, "Incremental contour-based topological segmentation for robot exploration," in *Proc. IEEE Int. Conf. Robot. Automat.*, Jul. 2017, pp. 2554–2561, doi: [10.1109/ICRA.2017.7989297](https://doi.org/10.1109/ICRA.2017.7989297).
- [13] L. Li et al., "Reconstruction of three-dimensional (3D) indoor interiors with multiple stories via comprehensive segmentation," *Remote Sens.*, vol. 10, no. 8, 2018, Art. no. 1281, doi: [10.3390/rs10081281](https://doi.org/10.3390/rs10081281).
- [14] B. Xiong, Y. Jin, F. Li, Y. Chen, Y. Zou, and Z. Zhou, "Knowledge-driven inference for automatic reconstruction of indoor detailed as-built BIMs from laser scanning data," *Automat. Construction*, vol. 156, Dec. 2023, Art. no. 105097, doi: [10.1016/j.autcon.2023.105097](https://doi.org/10.1016/j.autcon.2023.105097).
- [15] M. Mehranfar, A. Braun, and A. Borrmann, "From dense point clouds to semantic digital models: End-to-end AI-based automation procedure for Manhattan-world structures," *Automat. Construction*, vol. 162, Jun. 2024, Art. no. 105392, doi: [10.1016/j.autcon.2024.105392](https://doi.org/10.1016/j.autcon.2024.105392).
- [16] C. Mura, O. Mattausch, A. Jaspe Villanueva, E. Gobbetti, and R. Pajarola, "Automatic room detection and reconstruction in cluttered indoor environments with complex room layouts," *Comput. Graph.*, vol. 44, no. 1, pp. 20–32, 2014, doi: [10.1016/j.cag.2014.07.005](https://doi.org/10.1016/j.cag.2014.07.005).
- [17] Z. He, H. Sun, J. Hou, Y. Ha, and S. Schwertfeger, "Hierarchical topometric representation of 3D robotic maps," *Auton. Robots*, vol. 45, no. 5, pp. 755–771, Jun. 2021, doi: [10.1007/s10514-021-09991-8](https://doi.org/10.1007/s10514-021-09991-8).
- [18] E. Turner, P. Cheng, and A. Zakhor, "Fast, automated, scalable generation of textured 3D models of indoor environments," *IEEE J. Sel. Top. Signal Process.*, vol. 9, no. 3, pp. 409–421, Apr. 2015, doi: [10.1109/JSTSP.2014.2381153](https://doi.org/10.1109/JSTSP.2014.2381153).
- [19] Y. Cai and L. Fan, "An efficient approach to automatic construction of 3D watertight geometry of buildings using point clouds," *Remote Sens.*, vol. 13, no. 10, May 2021, Art. no. 1947, doi: [10.3390/rs13101947](https://doi.org/10.3390/rs13101947).
- [20] H. Macher, T. Landes, and P. Grussenmeyer, "From point clouds to building information models: 3D semi-automatic reconstruction of indoors of existing buildings," *Appl. Sci.*, vol. 7, no. 10, pp. 1–30, 2017, doi: [10.3390/app7101030](https://doi.org/10.3390/app7101030).
- [21] S. Nikoohemat, M. Peter, S. O. Elberink, and G. Vosselman, "Semantic interpretation of mobile laser scanner point clouds in Indoor scenes using trajectories," *Remote Sens.*, vol. 10, no. 11, Nov. 2018, Art. no. 1754, doi: [10.3390/rs10111754](https://doi.org/10.3390/rs10111754).
- [22] S. Nikoohemat, A. A. Diakité, S. Zlatanova, and G. Vosselman, "Indoor 3D reconstruction from point clouds for optimal routing in complex buildings to support disaster management," *Automat. Construction*, vol. 113, 2020, Art. no. 103109, doi: [10.1016/j.autcon.2020.103109](https://doi.org/10.1016/j.autcon.2020.103109).
- [23] R. Bormann, F. Jordan, W. Li, J. Hampp, and M. Hägele, "Room segmentation: Survey, implementation, and analysis," in *Proc. IEEE Int. Conf. Robot. Automat.*, Jun. 2016, pp. 1019–1026, doi: [10.1109/ICRA.2016.7487234](https://doi.org/10.1109/ICRA.2016.7487234).
- [24] R. Ambrus, S. Claici, and A. Wendt, "Automatic room segmentation from unstructured 3-D data of indoor environments," *IEEE Robot. Automat. Lett.*, vol. 2, no. 2, pp. 749–756, Apr. 2017.
- [25] A. Abdollahi, H. Arefi, S. Malihi, and M. Maboudi, "Progressive model-driven approach for 3D modeling of indoor spaces," *Sensors*, vol. 23, no. 13, Jun. 2023, Art. no. 5934, doi: [10.3390/s23135934](https://doi.org/10.3390/s23135934).
- [26] J. Jung, C. Stachniss, and C. Kim, "Automatic room segmentation of 3D laser data using morphological processing," *ISPRS Int. J. Geo-Inf.*, vol. 6, no. 7, Jul. 2017, Art. no. 206, doi: [10.3390/ijgi6070206](https://doi.org/10.3390/ijgi6070206).

- [27] J. Jung, C. Stachniss, S. Ju, and J. Heo, "Automated 3D volumetric reconstruction of multiple-room building interiors for as-built BIM," *Adv. Eng. Inform.*, vol. 38, pp. 811–825, Oct. 2018, doi: [10.1016/j.aei.2018.10.007](https://doi.org/10.1016/j.aei.2018.10.007).
- [28] F. Yang et al., "Semantic decomposition and recognition of indoor spaces with structural constraints for 3D indoor modelling," *Automat. Construction*, vol. 106, 2019, Art. no. 102913, doi: [10.1016/j.autcon.2019.102913](https://doi.org/10.1016/j.autcon.2019.102913).
- [29] K. Wu, W. Shi, and W. Ahmed, "Grid-based approach for the segmentation of multiple rooms from unstructured indoor point clouds," *J. Appl. Remote Sens.*, vol. 15, no. 04, Nov. 2021, Art. no. 044516, doi: [10.1117/1.jrs.15.044516](https://doi.org/10.1117/1.jrs.15.044516).
- [30] J. Yang, Z. Kang, L. Zeng, P. Hope Akwensi, and M. Sester, "Semantics-guided reconstruction of indoor navigation elements from 3D colorized points," *ISPRS J. Photogrammetry Remote Sens.*, vol. 173, pp. 238–261, Mar. 2021, doi: [10.1016/j.isprsjprs.2021.01.013](https://doi.org/10.1016/j.isprsjprs.2021.01.013).
- [31] S. Tang, X. Li, X. Zheng, B. Wu, W. Wang, and Y. Zhang, "BIM generation from 3D point clouds by combining 3D deep learning and improved morphological approach," *Automat. Construction*, vol. 141, Sep. 2022, Art. no. 104422, doi: [10.1016/j.autcon.2022.104422](https://doi.org/10.1016/j.autcon.2022.104422).
- [32] J. Martens and J. Blankenbach, "VOX2BIM+—A fast and robust approach for automated indoor point cloud segmentation and building model generation," *PFG - J. Photogrammetry, Remote Sens. Geoinformation Sci.*, vol. 1, pp. 1–22, May 2023, doi: [10.1007/s41064-023-00243-1](https://doi.org/10.1007/s41064-023-00243-1).
- [33] P. Hübner, M. Weinmann, S. Wursthorn, and S. Hinz, "Automatic voxel-based 3D indoor reconstruction and room partitioning from triangle meshes," *ISPRS J. Photogrammetry Remote Sens.*, vol. 181, pp. 254–278, Nov. 2021, doi: [10.1016/j.isprsjprs.2021.07.002](https://doi.org/10.1016/j.isprsjprs.2021.07.002).
- [34] I. Armeni et al., "3D semantic parsing of large-scale indoor spaces," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 1534–1543, doi: [10.1109/CVPR.2016.170](https://doi.org/10.1109/CVPR.2016.170).
- [35] D. Bobkov, M. Kiechle, S. Hilsenbeck, and E. Steinbach, "Room segmentation in 3D point clouds using anisotropic potential fields," in *Proc. IEEE Int. Conf. Multimedia Expo.*, Aug. 2017, pp. 727–732, doi: [10.1109/ICME.2017.8019484](https://doi.org/10.1109/ICME.2017.8019484).
- [36] A. Elseicy, S. Nikoohemat, M. Peter, and S. O. Elberink, "Space subdivision of indoor mobile laser scanning data based on the scanner trajectory," *Remote Sens.*, vol. 10, no. 11, pp. 1–26, 2018, doi: [10.3390/rs10111815](https://doi.org/10.3390/rs10111815).
- [37] S. Ochmann, R. Vock, and R. Klein, "Automatic reconstruction of fully volumetric 3D building models from oriented point clouds," *ISPRS J. Photogrammetry Remote Sens.*, vol. 151, pp. 251–262, 2019, doi: [10.1016/j.isprsjprs.2019.03.017](https://doi.org/10.1016/j.isprsjprs.2019.03.017).
- [38] S. Ochmann, R. Vock, R. Wessel, and R. Klein, "Automatic reconstruction of parametric building models from indoor point clouds," *Comput. Graph.*, vol. 54, pp. 94–103, 2016, doi: [10.1016/j.cag.2015.07.008](https://doi.org/10.1016/j.cag.2015.07.008).
- [39] Y. Cui, Q. Li, B. Yang, W. Xiao, C. Chen, and Z. Dong, "Automatic 3-D reconstruction of indoor environment with mobile laser scanning point clouds," *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.*, vol. 12, no. 8, pp. 3117–3130, Aug. 2019.
- [40] F. Yang, M. Che, X. Zuo, L. Li, J. Zhang, and C. Zhang, "Volumetric representation and sphere packing of indoor space for three-dimensional room segmentation," *ISPRS Int. J. Geo-Inf.*, vol. 10, 2021, Art. no. 739.
- [41] G. Liu, S. Wei, S. Zhong, S. Huang, and R. Zhong, "Reconstruction of indoor navigation elements for point cloud of buildings with occlusions and openings by wall segment restoration from indoor context labeling," *Remote Sens.*, vol. 14, no. 17, Aug. 2022, Art. no. 4275, doi: [10.3390/rs14174275](https://doi.org/10.3390/rs14174275).
- [42] Y. Lin, C. Wang, D. Zhai, W. Li, and J. Li, "Toward better boundary preserved supervoxel segmentation for 3D point clouds," *ISPRS J. Photogrammetry Remote Sens.*, vol. 143, pp. 39–47, Sep. 2018, doi: [10.1016/j.isprsjprs.2018.05.004](https://doi.org/10.1016/j.isprsjprs.2018.05.004).
- [43] R. Cai, H. Li, J. Xie, and X. Jin, "Accurate floorplan reconstruction using geometric priors," *Comput. Graph.*, vol. 102, pp. 360–369, Feb. 2022, doi: [10.1016/j.cag.2021.10.011](https://doi.org/10.1016/j.cag.2021.10.011).
- [44] R. Grompone Von Gioi, J. Jakubowicz, J.-M. Morel, and G. Randall, "LSD: A fast line segment detector with a false detection control," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 4, pp. 722–732, Apr. 2010, doi: [10.1109/TPAMI.2008.300](https://doi.org/10.1109/TPAMI.2008.300).
- [45] I. Suárez, J. M. Buenaposada, and L. Baumela, "ELSEDE: Enhanced line SEgment drawing," *Pattern Recognit.*, vol. 127, Jul. 2022, Art. no. 108619, doi: [10.1016/j.patcog.2022.108619](https://doi.org/10.1016/j.patcog.2022.108619).
- [46] C. Akinlar and C. Topal, "EDLines: A real-time line segment detector with a false detection control," *Pattern Recognit. Lett.*, vol. 32, no. 13, pp. 1633–1642, Oct. 2011, doi: [10.1016/j.patrec.2011.06.001](https://doi.org/10.1016/j.patrec.2011.06.001).
- [47] B. Okorn, X. Xiong, B. Akinci, and D. Huber, "Toward automated modeling of floor plans," in *Proc. Symp. 3D Data Process., Visual. Transmiss.*, 18–20, May 2010, vol. 2, [Online]. Available: [https://www.ri.cmu.edu/pub_files/2010/5/20093DPVTplanviewmodelingv13\(resubmitted\).pdf](https://www.ri.cmu.edu/pub_files/2010/5/20093DPVTplanviewmodelingv13(resubmitted).pdf)
- [48] K. Khoshelham, L. D. Vilariño, M. Peter, Z. Kang, and D. Acharya, "The ISPRS benchmark on indoor modelling," *Int. Arch. Photogrammetry, Remote Sens. Spatial Inf. Sci. - ISPRS Arch.*, vol. 42, no. 2W7, pp. 367–372, Sep. 2017, doi: [10.5194/isprs-archives-XLII-2-W7-367-2017](https://doi.org/10.5194/isprs-archives-XLII-2-W7-367-2017).
- [49] F. Li, W. Shi, Y. Tu, and H. Zhang, "Automated methods for indoor point cloud preprocessing: Coordinate frame reorientation and building exterior removal," *J. Building Eng.*, vol. 76, Oct. 2023, Art. no. 107270, doi: [10.1016/j.job.2023.107270](https://doi.org/10.1016/j.job.2023.107270).