# Building Component Detection on Unstructured 3D Indoor Point Clouds Using RANSAC-Based Region Growing 

Sangmin Oh ${ }^{1 \times D}$, Dongmin Lee ${ }^{2}{ }^{(D)}$, Minju Kim ${ }^{1}{ }^{(\mathbb{D}}$, Taehoon Kim ${ }^{1}$ and Hunhee Cho ${ }^{1, *}$<br>1 School of Civil, Environmental and Architectural Engineering, Korea University, 145 Anam-ro, Seongbuk-gu, Seoul 02841, Korea; timgemk47@korea.ac.kr (S.O.); minju830@korea.ac.kr (M.K.); kimth0930@korea.ac.kr (T.K.)<br>2 Department of Civil and Environmental Engineering, University of Michigan, 2350 Hayward St. G.G Brown Bldg., Ann Arbor, MI 48109, USA; dongminl@umich.edu<br>* Correspondence: hhcho@korea.ac.kr; Tel.: +82-2-3290-3328

Citation: Oh, S.; Lee, D.; Kim, M.; Kim, T.; Cho, H. Building Component Detection on Unstructured 3D Indoor Point Clouds Using RANSAC-Based Region Growing. Remote Sens. 2021, 13, 161. https://doi.org/10.3390/ rs13020161

Received: 20 November 2020
Accepted: 31 December 2020
Published: 6 January 2021

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


Copyright: © 2021 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/).


#### Abstract

With the advancement of light detection and ranging (LiDAR) technology, the mobile laser scanner (MLS) has been regarded as an important technology to collect geometric representations of the indoor environment. In particular, methods for detecting indoor objects from indoor point cloud data (PCD) captured through MLS have thus far been developed based on the trajectory of MLS. However, the existing methods have a limitation on applying to an indoor environment where the building components made by concrete impede obtaining the information of trajectory. Thus, this study aims to propose a building component detection algorithm for MLS-based indoor PCD without trajectory using random sample consensus (RANSAC)-based region growth. The proposed algorithm used the RANSAC and region growing to overcome the low accuracy and uniformity of MLS caused by the movement of LiDAR. This study ensures over $90 \%$ precision, recall, and proper segmentation rate of building component detection by testing the algorithm using the indoor PCD. The result of the case study shows that the proposed algorithm opens the possibility of accurately detecting interior objects from indoor PCD without trajectory information of MLS.


Keywords: building component detection; indoor point cloud; mobile laser scanner; random sample consensus; region growing

## 1. Introduction

In recent decades, light detection and ranging (LiDAR) has been used for obtaining a three-dimensional (3D) geometric representation of the indoor environment in the architectural, engineering, and construction (AEC) industry [1-3]. The LiDAR-based geometric representation indicates the shape descriptors of indoor building components by a format of point cloud data (PCD) [4,5]. Given the high speed and accuracy of LiDAR, methods for deriving the information of building components (e.g., structural member, non-structural member, and furniture) from the indoor PCD are developed to secure the effectiveness of facility management [6-8]. The information includes the location and size of mechanical, electrical, and plumbing [9-11], building components [12-14], and furniture in 3D space [15-17]. In the case of an indoor environment, the 3D scanning method for collecting PCD of the object as many as possible is selected for minimizing the occlusion caused by numerous objects $[18,19]$. The mobile laser scanner (MLS) has been widely used for scanning the indoor environment, which minimizes interference by moving the LiDAR.

The existing methods for deriving the information of building components from indoor PCD detect and segment the components based on the trajectory information of MLS. Considering the non-rigid shape of the indoor environment, the existence of openings (e.g., door, and window) is used for properly separating adjacent rooms using the trajectory of MLS [20-23]. In the process of deriving building components, the proper separation of room decreases the error on the segmentation of the inner wall's PCD [4,24,25]. Since the
adjacent rooms share an inner wall whose thickness is thinner than the vertical structural member, the existing method uses the LiDAR-oriented normal vector to minimize the error. The normal vector-based segmentation divides the under-segmented inner wall into two clusters that are included in different rooms respectively [26].

However, the existing methods have a limitation on applying to an environment that restricts the acquisition of trajectory information of MLS. In particular, the indoor environment has building components made of concrete, which makes it hard to use navigation systems that provide the real-time trajectory. Thus, MLS's trajectory in building interior is estimated by inversely calculating from the moving direction and distance obtained by the inertial measurement unit [27-29]. The approximate trajectory causes the error in detecting building components [30,31]. If the number of scanning positions increases, it has limitations on applying the existing methods to PCD without additional sensors because the accumulated error in PCD causes confusion of detecting the existence of building components based on the existing method suitable for high-quality PCD [32-34].

In this regard, this study proposes a building component detection algorithm that does not need trajectory information of MLS by using random sample consensus (RANSAC)based region growing for low-quality indoor PCD captured through MLS. This study is organized as follows: In Section 2, a literature review is conducted to derive the limitations of existing object detection methods for building components in indoor PCD. In Section 3, a building component detection algorithm is proposed that improves the limitation of existing object detection methods. In Section 4, a case study is conducted to verify the algorithm, and in Section 5, the proposed method is verified by analyzing the result of the case study. In Section 6, our conclusions are presented.

## 2. Related Work

### 2.1. Object Detection on Indoor PCD

Object detection on indoor PCD is conducted by localizing objects based on their feature descriptors (e.g., planar, cylindrical, and cone shape) followed by segmenting the clutter that belongs to a corresponding descriptor. This detection is used in many fields, especially the AEC industry to recognize obstacles (e.g., building components, doors, furniture, and stairs). For example, the path optimization for pedestrians and the assessment of structural members are conducted using recognized objects of indoor environments [35,36]. To accurately detect objects from indoor PCD that includes various objects, highly accurate PCD is needed because it is difficult to detect closed clutters such as the inner wall sharing adjacent rooms [21]. Furthermore, the density of points including the detected object must be uniform since a segmentation of the object is conducted by comparing it with a constant uniformity model to ensure the accuracy of geometric representation [37]. However, indoor PCD acquired through MLS has low accuracy and uniformity because of the LiDAR movement, which requires manipulation of PCD to suit object detection through pre-processing (e.g., segmentation, rotation, transferring, and labeling) [38]. Accordingly, this study analyzes the limitations of existing object detection caused by low accuracy and uniformity MLS, and derives why applying existing object detection methods is difficult. Finally, this study presents a measure to solve these difficulties.

### 2.2. Limitation on the Accuracy of MLS

LiDAR captures PCD by converting the flight time of light from the laser to the object surface into a distance (Figure 1a). In the case of indoor 3D scanning, the PCDs captured from one scanning are aligned according to the LiDAR's origin (Figure 1b). The aligned indoor PCD captured by an MLS has an error because the movement of origin is ignored (Figure 1c) [39]. Furthermore, the PCD acquisition for continuous 3D scanning on a large-scale environment accumulates the error in an integrated PCD.


Figure 1. (a) 3D scanning by LiDAR at $\theta=0^{\circ}$; (b) 3D scanning by LiDAR at $\theta=180^{\circ}$; (c) Integration error of point captured at $\theta=180^{\circ}$. LiDAR: light detection and ranging.

The accumulated error provides a thickness of planar objects, decreasing the accuracy of object detection by comparing clutters with predefined models. Since the models for comparing with clutters need uniformly distributed points on their surface, the thickness of the object decreases the accuracy of object detection. The existing methods transform the raw PCD into a suitable format for object detection to ensure accuracy [40]. However, it is hard to provide a proper format for the PCD that has a non-rigid shape, such as in an indoor environment. Thus, this study reviews the literature about the Manhattan-world assumption to transform the raw indoor PCD into a suitable format for object detection. Also, the literature of RANSAC is surveyed to solve the accuracy limitation using an MLS because of clutter with irregular thickness.

### 2.2.1. Manhattan-World Assumption

The Manhattan-world assumption is a hypothesis that defines the locational characteristics between objects that exist in a building [41]. The assumption regarding indoor PCD is that indoor building components (e.g., walls, floors, and ceilings) are mostly composed of horizontal components (i.e., floors and ceilings) and vertical components (i.e., walls and columns) [42]. This assumption yields effective analysis because it uses the topological feature to simplify the PCD acquired from the AEC industry whose size is larger than that of existing data from small objects (e.g., rabbits and statues).

However, the Manhattan-world assumption regarding indoor PCD can classify building components based on the angle between the normal vector of individual components and the ground after recognizing that an object exists based on the ground in the PCD. In other words, using the Manhattan-world assumption is difficult in object detection with indoor PCD that only has geometric data. Thus, this study segments points into components through aligning the components into planes (i.e., an XY-, XZ-, or YZ-plane) based on the normal vector. The raw indoor PCD is transformed into suitable formats of data using the Manhattan-world assumption.

### 2.2.2. RANSAC

RANSAC is a detection method for unstructured clutters that exist in the PCD by comparing the prior input model and clutters [43]. It selects points that are similar to the model's shape out of all points in the PCD as the RANSAC result, and all other unselected points are classified as outliers. To apply RANSAC, the number of points ( N ) and the shape of the model for comparison are set. Subsequently, the RANSAC conducts processes to change the locations of the model and to derive the least gross error of distance among the points of the clutter with the model (e.g., lines, curves, planar surfaces, and curved surfaces).

The RANSAC result is similar to that of regression. The difference is that regression finds the optimal line for all points that exist in the data, whereas RANSAC sets N for the detection of objects and removes unnecessary outliers to optimize the result (Figure 2). In particular, if more than two objects exist in the PCD, a threshold of N is set to ensure the efficiency and accuracy of object detection. Because RANSAC is used to derive points with
the least gross error based on the feature descriptors of the model regardless of the PCD density, it can be highly applicable when the object type and number are limited [44-46]. For example, indoor PCD has limited simple shapes such as planes, cylinders, and boxes as building components among objects, which is highly applicable to RANSAC [47,48].


Figure 2. Comparison regression line with the result of RANSAC. RANSAC: random sample consensus.

However, when the number of objects is over two, the number of objects that satisfy the prior inputted N for RANSAC-based detection can be larger than the actual number of objects if RANSAC is applied to all PCD. In particular, the curved part of connected planes causes the wrong detected objects concerning the points located in the part. Figure $3 \mathrm{a}, \mathrm{b}$ show examples of RANSAC-applied objects that exist in indoor PCD, in which gray points refer to raw PCD, and blue points refer to segmented points by RANSAC. Figure 3a shows the PCD with one object presented. Although the actual object is recognized by RANSAC based on pre-inputted N, many points are classified as outliers. Figure 3b shows fewer segmented blue points than Figure 3a because of the shared portion of objects in the edge [49]. To solve this problem, this study applied RANSAC to the segmented PCD, which was separated into each plane of the coordinate space based on the Manhattan-world assumption to minimize the error of building components detection.


Figure 3. (a) RANSAC on one object; (b) RANSAC on three objects.

### 2.3. Limitation on the Uniformity of MLS

The performance of LiDAR is determined by the number of lasers inside it. As the number of lasers increases, the scannable angle in the vertical direction ( $\theta$ in Figure 4 a ) and the scannable area (Figure 4b) are increasing. However, there were unscanned surfaces among the object surfaces as shown in Figure 4b, and the number of points may differ
even for the planar surfaces of the same area. In addition, the distance between points increases as the distance between the LiDAR and object increases, resulting in an increased unscanned surface ratio compared to the scanned surface. Thus, the density of points on the object surface differs in an environment where the distance between LiDAR and object changes according to the LiDAR location.


Figure 4. (a) Scanning process of LiDAR; (b) Scanned surface of object.
This characteristic degrades the uniformity of the indoor PCD acquired through an MLS, reducing the accuracy of object detection using a feature descriptor suitable for a constant density PCD [50]. To solve this problem, 3D scanning is conducted while moving a scanner in the $x-, y$-, and $z$-axes. However, it is difficult to acquire points on the surfaces of all objects at the same density. Thus, this study conducted a literature review of space decomposition and region growth to solve the aforementioned problem of an uneven PCD.

### 2.3.1. Space Decomposition

Space decomposition simplifies 3D PCD (w) into regular size box (Figure 5b) in which each box with points is called a voxel (Figure 5c). PCD acquired from large objects such as the indoors have numerous points, requiring excessive time for data processing if object detection is conducted on all points [51,52]. Thus, PCD with many planar objects (e.g., walls, floors, and ceilings) such as the case with indoor environments have a low accuracy reduction rate for data through data decomposition, which is why voxels are mainly used in object detection [53-55]. However, if voxels were produced according to whether points exist, a voxel with many points and one created by outlier would be regarded as the same point. For example, the space decomposition on indoor PCD where outlier frequently occurs by openings (e.g., windows, doors, and mirrors) degrades the accuracy of object detection [56]. Thus, this study aims to minimize the effect of outlier through individual data processing on voxels created through space decomposition.


Figure 5. (a) Raw PCD; (b) Space decomposition; (c) Voxels. PCD: Point Cloud Data.

### 2.3.2. Region Growing

Region growing aggregates points into larger regions to detect planar objects that exist in the PCD. As shown in Figure 6, points are connected based on preset criteria (e.g., distance and normal vector) starting from seed points or regions that exist in the PCD [56]. Region growing in 2D PCD combines points positioned closer than the preset distance based on points that belong to the seed region [57]. However, region growing in 3D PCD is conducted based on both the distance and the point's normal vector because many object shapes in 3D PCD are not planes but rather 3D curves [58,59]. This can be applied to all cases when the objects' surfaces are smooth and when the distance between points does not exceed the threshold of distance.


Figure 6. Region growing.
However, because points that share the same object surface are combined based on the normal vector, accurate normal vectors are required for individual points. In particular, because MLS cannot provide accurate normal estimation when there is no trajectory information, it is difficult to use region growing based on normal vectors [60,61]. Accordingly, the present study aims to minimize the accuracy reduction of object detection through directly applying region growing to voxels.

## 3. Proposed Algorithm

This study conducted a literature review of object detection in the case of indoor PCD acquired through an MLS to verify that existing methods have limitations on applying to that PCD. Moreover, the alternatives for indoor PCD are derived through a literature review of MLS trajectory-based building component detection to propose a suitable algorithm for indoor PCD where trajectory information is not available. The algorithm in this study applies RANSAC to voxels based on the space decomposition of indoor PCD to remove a thickness of planar objects. In addition, region growing is applied to conduct building component detection where the lack of uniformity is solved by RANSAC.

### 3.1. Overview

There are four steps in the algorithm: pre-processing, seed region generation, region growing, and building component detection. Each step is a data process on indoor PCD captured through an MLS. The pre-processing step has the following substeps: alignment the raw PCD with the coordinate planes, normal estimation based on the k-nearest neighbor
(k-NN) algorithm, PCD segmentation based on the normal vectors of points, and voxel generation by applying space decomposition on PCD. In the next region growing step, component candidates are made of connected regions through checking the connectivity between the seed region with the surrounding voxels. In the pre-processing and region growing steps, outliers are detected by applying RANSAC on voxels based on the inputted number of points. Finally, outlier is removed in the building component detection step based on the number of points, and building component detection is conducted through the normal vectors of regions without outlier.

### 3.2. Pre-Processing

As described, the pre-processing step has four substeps: alignment, normal estimation, PCD segmentation, and space decomposition. The raw indoor PCD captured through MLS without additional sensors which can provide the location information of MLS needs manual alignment with coordinate planes (i.e., XY, YZ, and XZ-plane). In this study, the raw PCD is rotated through manually aligning in CloudCompare to maximize the orthogonal feature of indoor environment as defined in the Manhattan-world assumption: the indoor environment has planar objects which are perpendicular or parallel with others [41]. In the normal estimation, the normal vectors of points are calculated based on the k-NN algorithm. The normal estimation based on $\mathrm{k}-\mathrm{NN}$ algorithm derives the normal vector calculating a mean value of cross products concerning the $k$-th adjacent points located in planar surface [62]. In fact, the planar surface derived from k-NN differs from the real object. However, the PCD used in this study has no LiDAR trajectory; thus, the normal vector indirectly estimated by the k-NN algorithm is used. To create a planar surface that belongs to clutter, k points that are close to the target point are grouped into one, and a 3D planar surface is derived as a planar surface whose normal vector is mean value of cross product among target point $k$ points. After this, PCD is segmented based on the normal vector of each point. Please note that the normal vector calculated through $\mathrm{k}-\mathrm{NN}$ is rounded with an infinite decimal; thus, the rounded normal vector to two decimal places identifies the tendency of the normal vector. This is to use the characteristic that most points existing in indoor PCD belong to a building component, as shown in Figure 7 [42]. Candidates of possible building components are segmented by the normal vector, as shown in Figure 7a-c, using the fact that indoor building components are either horizontal or vertical to each other based on the Manhattan-world assumption [41]. PCDs of Figure 7a-c are points that have the normal vector perpendicular to the YZ-plane, XZ-plane, and XYplane, respectively. This process is applied to minimize the error when applying RANSAC to many building components.


Figure 7. Segmentation based on the normal vectors. (a) PCD whose normal vector of x is one; (b) PCD whose normal vector of $y$ is one; (c) PCD whose normal vector of $z$ is one.

After completing the segmentation of raw PCD based on the normal vector, space decomposition is conducted to build voxels where RANSAC will apply. This process creates voxels including the minimum points required for RANSAC, through which it prevents applying RANSAC to outliers that are included in objects. The required variables in space decomposition in this study are the edge length of voxel (E in Figure 8a) and the minimum number of points inside a voxel to distinguish outliers ( N in Figure 8b). The edge length of the voxel is related to the rate of PCD simplification. The longer the edge length, the simpler the PCD. However, this may cause a problem where the indoor features decrease. Thus, this study sets the length of edge between the maximum and minimum values of the LiDAR error. In addition, the minimum number of points inside a voxel is set to distinguish between points on the planar surface of the actual object and outliers. The number is set to three in this study, which is the minimum number of points that can make a planar surface. If the number of points selected through RANSAC based on the degree for filtering outlier, as shown in the selected points in Figure 8b, is more than three, the voxel is recognized as a region.


Figure 8. (a) Point in voxel; (b) Selected point colored by blue and noise colored by red.

### 3.3. Seed Region Generation

Seed region generation is a process of inputting the selected points in Figure 8b into a seed region after selecting a voxel without outliers. The voxel excluding outliers is a voxel whose N is greater than three. This process filters points that have a large difference with the normal vector of the plane where the voxel is present (Figure 8b). Based on the parameter of difference regarding the normal vector, the algorithm calculates the included angle between the mean value of normal vector of points in the voxel with each point, and removes the point whose normal vector has larger degree than the parameter. Model fitting was conducted with all points (Figure 9a) including outliers. The result obtained by voxel-based RANSAC proposed in this study was similar to the surface of the object where the voxel was present, as shown in Figure 9c, in contrast with the previous RANSAC where an unsuitable plane fitting was made as shown in Figure 9b. The seed region of the raw PCD was derived by applying the voxel-based RANSAC, and outliers was removed to improve the accuracy of the region growing.

### 3.4. Region Growing

This study checks the connectivity between the seed region with adjacent n regions, as shown in Figure 10a, to apply region growing to seed region derived by RANSAC. However, each region has no shared points because of space decomposition. Thus, this study conducted connectivity checking on 26 adjacent regions around the seed region (the red region in Figure 10b). The 26 adjacent regions (the gray regions in Figure 10b) are the maximum number of regions that share edges, faces, or vertices with the seed region. First, adjacent regions are checked by determining whether points are present in adjacent regions. If points are present in regions, the connectivity checking is conducted among the seed
region and adjacent regions through comparison between the normal vector of points that exist inside regions. The normal vector of points located in the colored regions is extracted, and the mean value of each colored regions are compared to check the connectivity as shown in Figure 10c.


Figure 9. (a) Inputted PCD; (b) RANSAC on entire PCD; (c) RANSAC on voxels.


Figure 10. (a) Points in regions; (b) Seed region colored by red and adjacent regions colored by grey; (c) Process of connectivity checking.

### 3.5. Building Component Detection

The building component detection in this study first distinguishes planar objects created through a region growing into candidates of horizontal and vertical components (Figure 11a). Components are distinguished based on the normal vector (i.e., $X Y$-plane $=(0,0,1)$, XZ-plane $=(0,1,0)$, YZ-plane $=(1,0,0))$, and then a building component is classified through checking whether it overlaps with other candidates. In the process of classifying, the candidate that the number of points of it is lower than the predefined number of points for eliminating outliers is removed. Overlapping between candidates is determined by checking the points included in the candidates, as shown in Figure 11, thereby eliminating overlapping candidates (Figure 11b). If candidates satisfy the presented two criteria, they are classified as building components according to the normal vector.


Figure 11. (a) Candidates of structural members; (b) Overlap between candidates.

## 4. Case Study

This case study verifies the building component detection algorithm, detecting and segmenting building components from the PCD without trajectory information acquired through MLS. In particular, this study focuses on verifying the applicability of proposed algorithm for indoor PCD which has high error described as the thickness of planar surface caused by several reasons such as the absence of additional sensors and movement of MLS. Although the limitation on detecting building components can be improved by employing the trajectory information, this study proposes the novel approach based on the geometry data for detecting components from PCD regardless of quality of data (e.g., registration error, interference points, etc.).

This algorithm was implemented in MATLAB R2020a. First, the precision and recall of the building components are calculated after applying the algorithm to indoor PCD. The precision and recall used in this study are indicators of the recognition rate for this algorithm. The results were derived by setting three cases: true positive refers to the recognition of actual components, false positive refers to cases where the recognized component is not an actual component, and false negative refers to cases where an actual component is not recognized. The precision and recall of the result are calculated by the Equations (1) and (2) [63]. Next, the over- and under-segmentation rates of the recognized building components are derived. Over- and under-segmentation refer to cases where a single component is segmented into two or more components and where two or more components are combined into one, respectively, indicating the recognition precision [64].

$$
\begin{align*}
& \text { Precision }=\frac{\text { True Psitive }}{\text { True Positive }+ \text { False Positive }}  \tag{1}\\
& \text { Recall }=\frac{\text { True Psitive }}{\text { True Positive }+ \text { False Negative }} \tag{2}
\end{align*}
$$

### 4.1. Overview

The datasets in this case study were acquired through an MLS using HDL-32E and VLP-16. Table 1 presents the PCD's information, whose environment includes various object obstacles (e.g., tables, chairs, and whiteboards) that occlude building components. To verify the accuracy and applicability of the proposed algorithm, this case study used the different three datasets that have different shape, type, interior objects, and accuracy of PCD. In particular, the huge obstacle of segmentation for indoor PCD is non-rigid shape of it, thus this study employed the datasets whose shapes are definitely different. The number of rooms in each dataset are seven, four, and five, respectively. Furthermore, the Dataset \#2 and Dataset \#3 have spaces that are not the room: corridor and kitchen-integrated with living room. Through comparing the results of Dataset \#1 and \#2 with Dataset \#3, the applicability of this study would be verified for indoor PCDs that have non-rigid shape.

Table 1. Datasets of the case study.

|  | Dataset \#1 | Dataset \#2 | Dataset \#3 |
| :---: | :---: | :---: | :---: |
| Location | Goyang, South Korea | Seoul, South Korea | Pohang, South Korea |
| Points | 26 M | 70 M | 88 M |
| LiDAR | HDL-32E | VLP-16 | VLP-16 |

Next, the datasets that have different type and interior object of indoor environment were employed to test that the proposed algorithm could properly detect and segment the building components from the different conditions. Dataset \#1, \#2, and \#3 were captured from detached house, office, and apartment, respectively.

Lastly, the applicability for capacity of LiDAR will be verified through applying the proposed algorithm to two PCDs that have different density captured through HDL-32E and VLP-16. If the algorithm were able to ensure the accurate results of case study on different LiDAR, this study could be possible to extend the applicability for low-density indoor PCD.

The parameters used in this case study is presented in Table 2. The number of points for estimating normal vector based on $\mathrm{k}-\mathrm{NN}$ algorithm is 100 since the number is the maximum number of points calculated by capacity of hardware used in this study. Although the estimated normal vector is more similar if the number of points for calculating the normal vector was increasing, the excessive number of points causes the inaccurate normal vector since the points gathered by the number are regarded as one object. This study set the number of points 100, which can derive the approximate value of normal vector based on the iterative testing. In addition, the threshold for filtering outlier and the number of points for RANSAC are three. The three is minimum number of points for organizing the surface; thus, this study set the threshold three for filtering the outliers that are insufficient for creating the triangular plane. The degrees for filtering outlier in voxel and checking the connectivity between regions are two and five degrees. The degrees for filtering and checking are optimized value derived through iteratively testing the result of building component detection; however, the degree depends on the thickness of planar objects located in PCD. In particular, the LiDAR sensors used in the have different accuracy and thickness of PCD; thus, the degrees for checking of Dataset \#2 and \#3 are five since the VLP-16 has lower accuracy (i.e., $\pm 3 \mathrm{~cm}$ ) than the HDL-32E ( $\pm 2 \mathrm{~cm}$ ).

Table 2. Parameters of the case study.

| Parameters (unit) | Value |
| :---: | :---: |
| The number of points for k-NN | 100 |
| Threshold for filtering | 3 |
| The number of points for RANSAC, N | 3 |
| Degree for filtering $\left({ }^{\circ}\right)$ | 2 |
| Degree for checking in Dataset \#1 $\left({ }^{\circ}\right)$ | 3 |
| Degrees for checking in Dataset \#2 and \#3 $\left({ }^{\circ}\right)$ | 5 |

### 4.2. Result

Figures 12-14 presents the raw PCD and results of the building component detection from the datasets derived through the algorithm, and Tables 3-5 present the precision, recall, and over- and under-segmentation rates. Table 6 presents the running time for processing steps of Dataset \#1, \#2, and \#3.


Figure 12. Implementation results of Dataset \#1. (a) Raw PCD; (b) Top view of detected building components; (c) Front isometric view of detected building components.


Figure 13. Implementation results of Dataset \#2. (a) Raw PCD; (b) Top view of detected building components; (c) Front isometric view of detected building components.


Figure 14. Implementation results of Dataset \#3. (a) Raw PCD; (b) Top view of detected building components; (c) Front isometric view of detected building components.

Table 3. Experiment results of Dataset \#1.

|  | Wall | Ceiling | Floor |
| :---: | :---: | :---: | :---: |
| Detected/Real component | $39 / 41$ | $7 / 7$ | $2 / 1$ |
| Precision (\%) | 94.9 | 100.0 | 100.0 |
| Recall (\%) | 86.1 | 100.0 | 100.0 |
| Over-segmentation rate (\%) | 5.4 | 0.0 | 50.0 |
| Under-segmentation rate $(\%)$ | 0.0 | 0.0 | 0.0 |

Table 4. Experiment results of Dataset \#2.

|  | Wall | Ceiling | Floor |
| :---: | :---: | :---: | :---: |
| Detected/Real component | $53 / 51$ | $5 / 5$ | $3 / 1$ |
| Precision (\%) | 84.9 | 100.0 | 100.0 |
| Recall (\%) | 80.4 | 100.0 | 100.0 |
| Over-segmentation rate (\%) | 4.8 | 0.0 | 66.7 |
| Under-segmentation rate (\%) | 0.0 | 0.0 | 0.0 |

Table 5. Experiment results of Dataset \#3.

|  | Wall | Ceiling | Floor |
| :---: | :---: | :---: | :---: |
| Detected /Real component | $41 / 42$ | $4 / 5$ | $1 / 1$ |
| Precision (\%) | 95.1 | 100.0 | 100.0 |
| Recall (\%) | 88.6 | 80.0 | 100.0 |
| Over-segmentation rate (\%) | 5.1 | 0.0 | 0.0 |
| Under-segmentation rate (\%) | 0.0 | 0.0 | 0.0 |

Table 6. Running time for processing steps of Dataset \#1, \#2, and \#3. (Unit: seconds).

|  | Dataset \#1 | Dataset \#2 | Dataset \#3 |
| :---: | :---: | :---: | :---: |
| Pre-processing | 230.1 | 602.3 | 720.4 |
| Seed Region Generation | 12.5 | 15.8 | 17.2 |
| Region Growing | 125.8 | 294.4 | 330.5 |
| Building Component Detection | 23.9 | 30.5 | 28.2 |
| Total | 392.3 | 943.0 | 1096.3 |

The precision set in this study refers to the ratio of objects that correspond to building components, and the recall refers to the recognized ratio of actual building components that exist in the building. As the values are higher, the algorithm is used to recognize building components that exist indoors more accurately. As presented in Tables 3-5, the precision and recall values of the floor and ceiling in all datasets were all $100 \%$ except the ceiling of Dataset \#3; this was because there were few floors and ceilings in the building, and the interference caused by indoor objects in the building was lower for floors and ceilings than that for walls. In Sections 4.2.1-4.2.4, this study analyzed the results of horizontal components (i.e., floors and ceilings) and vertical components (i.e., walls) to verify the accuracy and applicability of this algorithm.

The running time for processing steps of Dataset \#1, \#2, and \#3 are presented in Table 6. The time-consuming step is pre-processing since the normal estimation using k-NN algorithm calculates the normal vector of all points. Furthermore, the time for normal estimation is proportional with the number of points in PCD; thus, the Dataset \#3 consumed the 514.2 seconds for estimating the normal vector based on the number of points for k-NN algorithm in MATLAB R2020a since the Dataset \#3 has largest number of points. Compared to the Dataset \#3, the cases of Dataset \#1 and \#2 consumed less time for estimating, 138.6 and 385.3 seconds, but the step take most portion of time for pre-processing. The next time-consuming step is region growing because the algorithm
iteratively checked all regions generated from the seed region generation step. Although the time is decreasing through minimizing the size of voxel since the time for region growing is affected by the size of voxel used in seed region growing, it is difficult to properly segment the PCD since the big size of voxel removes the topological feature of objects. The average rate of time for seed region generation and building components detection took 1.8 and $3.4 \%$ of total processing time. Compared with the pre-processing and region growing whose processing time depends on the number of points, the other steps depend on the size of voxel; thus, it can be optimized by manipulating the size.

### 4.2.1. Precision and Recall of Vertical Component Detection

The detection of walls in all datasets showed higher precision than recall. The average precision was $91.63 \%$, meaning that the proposed algorithm accurately detected components of the indoor PCD. Dataset \#1 was acquired from a house with seven rooms (Figure 15a). Figure 15b shows the comparison between the detected walls and the floor plan, and Figure 15c presents the yellow-color walls that were not detected. In particular, the area of No. 1 wall in Figure 15c was very narrow, so the number of points that existed in the voxel was significantly small, creating an error in the calculation of the normal vector based on the distance between points. The points in the No. 1 wall were removed through region growing, causing the non-detection. In addition, the No. 2 wall in Figure 15c was not detected because it was removed from PCD segmentation in pre-processing. However, most undetected components in Datasets \#2 and \#3 were columns inside the buildings, the same as was observed for the No. 1 wall in Figure 15c. These columns were very narrow and thus were not detected. Except for such special circumstances, the results verified that the precision of all walls exceeded $90 \%$.


Figure 15. (a) Detected walls; (b) Comparison walls with floorplan; (c) Undetected walls.

### 4.2.2. Segmentation of Vertical Component

The accurate segmentation of recognized components is also important because it is necessary to calculate the accurate dimension and location of the building components. In particular, the thickness of inner wall that shares two rooms is affected by their partition types (e.g., structural member, non-structural member, etc.), which contrasts with outer walls. PCD of inner walls acquired through MLS is expressed as two planes that are close to each other. Therefore, the inner walls that have little thickness may be segmented into a single plane. To prevent this under-segmentation, the size of voxel used in the RANSAC and region growing was optimized for separating the surfaces of inner wall into two rooms. This study verified that no under-segmentation occurred in all datasets. In contrast, over-segmentation occurred in all datasets because of wall segmentation by indoor curtains. In particular, Dataset \#2 had the segmentation of one side wall into many walls because curtains had been installed in all windows to minimize the LiDAR error caused by light from outside (Figure 16).


Figure 16. Over-segmentation of walls.

### 4.2.3. Precision and Recall of Horizontal Component Detection

Floors and ceilings, all of which were horizontal components, were fully detected in all datasets except for the ceiling in Dataset \#3, as verified in the precision and recall of the components. The reason for the undetected ceiling in Dataset \#3 was that points of the ceiling were not acquired because of the narrow ceiling space in the bathroom attached to the master bedroom. In particular, the ceiling was not scanned because of the interference of light emitted, while an MLS entered the master bedroom because of the curb that existed between the ceilings of the master bedroom and bathroom in the master bedroom (Figure 17). Except for this detection error caused by these characteristics, the results verified that the precision of the horizontal building components was high for all datasets.


Figure 17. Interference on the ceiling of the bathroom.

### 4.2.4. Segmentation of Horizontal Component

The over- and under-segmentation of horizontal building components in this study occurred because of excessive separation or combinations of floors and ceilings. In particular, ceilings were separated by the curb of the ceiling that separates each room, whereas floors in all rooms were connected. In this study, accurate detection was obtained when a single floor was accurately segmented in a single building. As presented in Tables 3-5, this algorithm accurately detected the ceilings in the segmentation of ceilings, whereas over-segmentation occurred in floors in Datasets \#1 and \#2. In particular, the objects inside the room interfered with the floors and LiDAR in Dataset \#2 (Figure 18).


Figure 18. Case of over-segmentation on floor.

## 5. Discussion

This study verified the building component detection algorithm through the case study of indoor PCD captured through MLS. The mean values of detection precision and recall secured over $92 \%$ using the proposed building component detection algorithm; in addition, the precision and recall of application whose result derived from International Society for Photogrammetry and Remote Sensing (ISPRS) benchmark datasets got over $95 \%$. The ISPRS benchmark datasets ae public datasets provided by Working Group IV 5. Based on the difference between the case of raw PCD and ISPRS datasets, it is found that the quality of PCD affects the result of detection since the ISPRS datasets has more uniform point density than PCD used in the case study; however, the proposed algorithm can recognize over $92 \%$ of building components from low-quality data without trajectory information through hierarchical steps. In particular, the detection error caused by the lack of normal vector was minimized through the process of the segmentation on indoor PCD, and the accuracy of the total process was ensured by filtering outliers in raw points, segmented clutter, and grown regions in the detection step. Furthermore, RANSAC was used for separating outlier that was not eliminated in either the pre-processing or seed region generation steps. The PCD without outlier could provide the more uniform feature descriptor, which improved the accuracy of detection.

The main parameter affecting the result of detection is degrees for checking since the parameter helps the algorithm properly connecting the adjacent regions. Compared with the degree for filtering that removes useless points, the parameter can connect among different planar surfaces or separate one surface into more than two surfaces. In particular, the algorithm divided the one surface into numerous planes if the parameter has low degree since the raw indoor PCD with high registration error generates more than two seed regions from planar surface. Furthermore, a high degree caused under-segmentation since the joint part located between adjacent planar building components has gradually changing normal vector from one of components to another; thus, the algorithm has trouble to decide whether the adjacent regions located in the joint part due to the similarity of normal vector. The next parameter is the number of points for $k-N N$ because the parameter affects the accuracy of normal estimation. Since the algorithm uses the degree calculated from normal vector, the accuracy of normal estimation affects the whole processes of the algorithm. However, the normal vector can be accurately estimated by k-NN algorithm for planar surface then the excessive points for normal estimation cause meaningless processing time. The parameter for $\mathrm{k}-\mathrm{NN}$ is commonly used in normal estimation for indoor PCD. To optimize the accuracy and efficiency of the algorithm, this study iteratively manipulated the parameter and set as presented in Table 2. In addition, the parameters-threshold for filtering and the number of points for RANSAC-are determined based on the minimum
number of points for generating the plane. The degree for filtering was set for removing the outliers located in voxel, which has same role in detecting outlier with the degree for checking; however, the parameter has limitation on removing the region because the normal vector of points located in same voxel is same if the number of points was lower than the number of points for $\mathrm{k}-\mathrm{NN}$. In this study, the parameter is used for removing the outliers located in the joint part.

Through applying the proposed algorithm, building component detection can be done on indoor PCD that has high registration error without trajectory information of MLS. The proposed algorithm provides three benefits to contribute to the body of knowledge in the AEC industry. First, the high accuracy of this algorithm makes it possible to derive 3D geometric representation from scanning data of an indoor environment without trajectory information of LiDAR. The proposed algorithm employs the Manhattan-world assumption to check overlap between the building component's position. The overlap increases the accuracy of derived building components in the detection step on an indoor PCD without trajectory of LiDAR. Second, the proposed algorithm secures the consistent accuracy of building component detection regardless of the quality of indoor PCD. The accuracy and proper segmentation rate of detection are over $80 \%$ in the case study using datasets captured through two kinds of LiDAR with different capacities. In particular, the result of building components detection on Dataset \#1 and \#3 prove that the proposed algorithm can detect components accurately from the low-quality indoor PCD. Consequently, the requirement for LiDAR has low relation with the accuracy of detection, which makes it possible to use low-cost LiDAR on building component detection. Lastly, the 3D geometric representation derived through the proposed algorithm can be used for supporting the accurate Building Information Modeling (BIM) generation with low-quality PCD. The executors who use the traditional method for generating BIM from PCD manually draw the lines in accordance with the structural members. The unstructured PCD emits the information of clutters for distinguishing them into structural member because it only has geometry data; thus, executors who generate as-built BIM must judge whether the clutter is structural members. In contrast, the proposed algorithm provides the existence and location of building components, which lets the executors choose a structural member from the properly segmented components. The candidates of structural member can improve the accuracy of the judgment since the segmentation of this algorithm removes the outlier that causes the false geometric representation of member.

Nonetheless, the proposed algorithm had difficulty to apply to non-aligned PCD whose building components are not aligned with the coordinate planes. To solve the limitation, the algorithm needs additional process for alignment of raw PCD. In addition, the comparison of the result derived from the case study with the previous studies is not conducted because it has limitations on applying the existing methods to our datasets with higher registration error than the ISPRS benchmark datasets [62-64]; thus, this study verified the algorithm about applicability of it for low-quality indoor PCD.

## 6. Conclusions

This study proposed a building component detection algorithm that was suitable for indoor PCD without MLS trajectory information by using the features of the indoor environment. The proposed algorithm was verified through the case study, in which acquiring MLS trajectory information was difficult. In particular, over $90 \%$ accuracy of building components detection was ensured by this algorithm from the low-quality raw PCD without the additional sensors. Furthermore, the building components detected by this algorithm has 3D geometric representation which can be used for reconstructing the interior environment. Through applying the proposed algorithm, we believed that the reverse engineering of the as-is indoor is possible using the indoor PCD without trajectory. For future study, 3D reconstruction algorithms for the indoors of buildings will be developed to segment the physically connected components regardless of condition of indoor environment.


#### Abstract

Author Contributions: S.O. designed and devised the algorithm of this study and conducted the case study and analyzed the result; M.K. collected the data; S.O., M.K., and D.L. wrote the paper; H.C. and T.K. offered advice on this study; H.C. edited the paper. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2018R1A4A1026027).

Institutional Review Board Statement: Not applicable. Informed Consent Statement: Not applicable. Conflicts of Interest: The authors declare no conflict of interest.


## References

1. Thomson, C.P.H. From Point Cloud to Building Information Model: Capturing and Processing Survey Data Towards Automation for High Quality 3D Models to Aid a BIM Process. Ph.D. Thesis, University College London, London, UK, 2016.
2. Rebolj, D.; Pučko, Z.; Babič, N.Č.; Bizjak, M.; Mongus, D. Point cloud quality requirements for Scan-vs-BIM based automated construction progress monitoring. Autom. Constr. 2017, 84, 323-334. [CrossRef]
3. Bueno, M.; Bosché, F.; González-Jorge, H.; Martínez-Sánchez, J.; Arias, P. 4-Plane congruent sets for automatic registration of as-is 3D point clouds with 3D BIM models. Autom. Constr. 2018, 89, 120-134. [CrossRef]
4. Pauwels, P.; Zhang, S.; Lee, Y.C. Semantic web technologies in AEC industry: A literature overview. Autom. Constr. 2017, 73,145-165. [CrossRef]
5. Staats, B.R.; Diakité, A.A.; Voûte, R.L.; Zlatanova, S. Detection of doors in a voxel model, derived from a point cloud and its scanner trajectory, to improve the segmentation of the walkable space. Int. J. Urban. Sci. 2019, 23, 369-390. [CrossRef]
6. Liao, L.; Teo, E.A.L. Managing critical drivers for building information modelling implementation in the Singapore construction industry: An organizational change perspective. Int. J. Constr. Manag. 2019, 19, 240-256. [CrossRef]
7. $\mathrm{Xu}, \mathrm{J} . ;$ Lu, W.; Xue, F.; Chen, K. ‘Cognitive facility management': Definition, system architecture, and example scenario. Autom. Constr. 2019, 107, 102922. [CrossRef]
8. Hilal, M.; Maqsood, T.; Abdekhodaee, A. A scientometric analysis of BIM studies in facilities management. Int. J. Build. Pathol. Adapt. 2019, 37, 122-139. [CrossRef]
9. Wang, J.; Wang, X.; Shou, W.; Chong, H.Y.; Guo, J. Building information modeling-based integration of MEP layout designs and constructability. Autom. Constr. 2016, 61, 134-146. [CrossRef]
10. Pärn, E.A.; Edwards, D.J. Conceptualising the FinDD API plug-in: A study of BIM-FM integration. Autom. Constr. 2017, 80, 11-21. [CrossRef]
11. Hu, Z.Z.; Tian, P.L.; Li, S.W.; Zhang, J.P. BIM-based integrated delivery technologies for intelligent MEP management in the operation and maintenance phase. Adv. Eng. Softw. 2018, 115, 1-16. [CrossRef]
12. Ramaji, I.J.; Memari, A.M. Interpretation of structural analytical models from the coordination view in building information models. Autom. Constr. 2018, 90, 117-133. [CrossRef]
13. Hasan, A.M.; Torky, A.A.; Rashed, Y.F. Geometrically accurate structural analysis models in BIM-centered software. Autom. Constr. 2019, 104, 299-321. [CrossRef]
14. Basta, A.; Serror, M.H.; Marzouk, M. A BIM-based framework for quantitative assessment of steel structure deconstructability. Autom. Constr. 2020, 111, 103064. [CrossRef]
15. Conde, A.J.L.; García-Sanz-Calcedo, J.; Rodríguez, A.M.R. Use of BIM with photogrammetry support in small construction projects. Case study for commercial franchises. J. Civ. Eng. Manag. 2020, 26, 513-523. [CrossRef]
16. Na, S.; Hong, S.W.; Jung, S.; Lee, J. Performance evaluation of building designs with BIM-based spatial patterns. Autom. Constr. 2020, 118, 103290. [CrossRef]
17. Sydora, C.; Stroulia, E. Rule-based compliance checking and generative design for building interiors using BIM. Autom. Constr. 2020, 120, 103368. [CrossRef]
18. Cavalliere, C.; Dell’Osso, G.R.; Favia, F.; Lovicario, M. BIM-based assessment metrics for the functional flexibility of building designs. Autom. Constr. 2019, 107, 102925. [CrossRef]
19. Omar, T.; Nehdi, M.L. Data acquisition technologies for construction progress tracking. Autom. Constr. 2016, 70, 143-155. [CrossRef]
20. Zheng, H. Recognizing Pole-Like Objects from Mobile LiDAR Data. Master's Thesis, University of Calgary, Calgary, AB, Canada, 2016.
21. Nikoohemat, S.; Peter, M.; Elberink, S.O.; Vosselman, G. Exploiting Indoor Mobile Laser Scanner Trajectories for Semantic Interpretation of Point Clouds. In ISPRS Annals of Photogrammetry, Remote Sensing \& Spatial Information Sciences 4, Proceedings of the ISPRS Geospatial Week, Wuhan, China, 18-22 September 2017; ISPRS: Hannover, Germany, 2017.
22. Liu, J.; Pu, J.; Sun, L.; He, Z. An approach to robust INS/UWB integrated positioning for autonomous indoor mobile robots. Sensors 2019, 19, 950. [CrossRef]
23. Luo, W.; Li, L. Automatic geometry measurement for curved ramps using inertial measurement unit and 3D LiDAR system. Autom. Constr. 2018, 94, 214-232. [CrossRef]
24. Nikoohemat, S.; Peter, M.; Oude Elberink, S.; Vosselman, G. Semantic interpretation of mobile laser scanner point clouds in indoor scenes using trajectories. Remote Sens. 2018, 10, 1754. [CrossRef]
25. Elseicy, A.; Nikoohemat, S.; Peter, M.; Elberink, S.O. Space subdivision of indoor mobile laser scanning data based on the scanner trajectory. Remote Sens. 2018, 10, 1815. [CrossRef]
26. Walczak, J.; Andrzejczak, G.; Scherer, R.; Wojciechowski, A. Normal Grouping Density Separation (NGDS): A Novel Object-Driven Indoor Point Cloud Partition Method. In International Conference on Computational Science; Springer: Cham, Germany, 2020; pp. 100-114.
27. Kumar, G.A.; Patil, A.K.; Patil, R.; Park, S.S.; Chai, Y.H. A LiDAR and IMU integrated indoor navigation system for UAVs and its application in real-time pipeline classification. Sensors 2017, 17, 1268. [CrossRef] [PubMed]
28. Sadruddin, H.; Mahmoud, A.; Atia, M.M. Enhancing Body-Mounted LiDAR SLAM using an IMU-based Pedestrian Dead Reckoning (PDR) Model. In Proceedings of the IEEE 63rd International Midwest Symposium on Circuits and Systems (MWSCAS), Springfield, MA, USA, 9-12 August 2020; pp. 901-904.
29. Karam, S.; Lehtola, V.; Vosselman, G. Strategies to Integrate IMU and LIDAR SLAM for Indoor Mapping. In ISPRS Annals of Photogrammetry, Remote Sensing \& Spatial Information Sciences; ISPRS: Hannover, Germany, 2020; Volume 5.
30. Li, H.; Wen, X.; Guo, H.; Yu, M. Research into Kinect/inertial measurement units based on indoor robots. Sensors 2018, 18, 839. [CrossRef] [PubMed]
31. Liu, X.; Zhang, L.; Qin, S.; Tian, D.; Ouyang, S.; Chen, C. Optimized LOAM Using Ground Plane Constraints and SegMatch-Based Loop Detection. Sensors 2019, 19, 5419. [CrossRef] [PubMed]
32. Truong-Hong, L.; Laefer, D.F. Quantitative evaluation strategies for urban 3D model generation from remote sensing data. Comput. Graph. 2015, 49, 82-91. [CrossRef]
33. Shi, W.; Ahmed, W.; Li, N.; Fan, W.; Xiang, H.; Wang, M. Semantic geometric modelling of unstructured indoor point cloud. ISPRS Int. J. Geo-Inf. 2019, 8, 9. [CrossRef]
34. Bassier, M.; Vergauwen, M.; Poux, F. Point Cloud vs. Mesh Features for Building Interior Classification. Remote Sens. 2020, 12, 2224. [CrossRef]
35. Wu, K.; Shi, W.; Ahmed, W. Structural Elements Detection and Reconstruction (SEDR): A Hybrid Approach for Modeling Complex Indoor Structures. ISPRS Int. J. Geo-Inf. 2020, 9, 760. [CrossRef]
36. Huang, H.C.; Hsieh, C.T.; Yeh, C.H. An indoor obstacle detection system using depth information and region growth. Sensors 2015, 15, 27116-27141. [CrossRef]
37. Chen, J.; Fang, Y.; Cho, Y.K. Performance evaluation of 3D descriptors for object recognition in construction applications. Autom. Constr. 2018, 86, 44-52. [CrossRef]
38. Huang, H.; Brenner, C.; Sester, M. A generative statistical approach to automatic 3D building roof reconstruction from laser scanning data. ISPRS J. Photogramm. Remote Sens. 2013, 79, 29-43. [CrossRef]
39. Mattausch, O.; Panozzo, D.; Mura, C.; Sorkine-Hornung, O.; Pajarola, R. Object detection and classification from large-scale cluttered indoor scans. In Computer Graphics Forum; John Wiley \& Sons Ltd.: Hoboken, NJ, USA, 2014; Volume 33, pp. 11-21.
40. He, L.; Jin, Z.; Gao, Z. De-Skewing LiDAR Scan for Refinement of Local Mapping. Sensors 2020, 20, 1846. [CrossRef] [PubMed]
41. Ma, L.; Li, Y.; Li, J.; Wang, C.; Wang, R.; Chapman, M.A. Mobile laser scanned point-clouds for road object detection and extraction: A review. Remote Sens. 2018, 10, 1531. [CrossRef]
42. Coughlan, J.M.; Yuille, A.L. Manhattan world: Compass direction from a single image by bayesian inference. In Proceedings of the Seventh IEEE International Conference on Computer Vision, Kerkyra, Greece, $20-27$ September 1999; Volume 2, pp. 941-947.
43. Budroni, A.; Böhm, J. Automatic 3D modelling of indoor manhattan-world scenes from laser data. In Proceedings of the International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, Newcastle upon Tyne, UK, 21-24 June 2010; pp. 115-120.
44. Fischler, M.A.; Bolles, R.C. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. Comтип. ACM 1981, 24, 381-395. [CrossRef]
45. Grilli, E.; Menna, F.; Remondino, F. A review of point clouds segmentation and classification algorithms. Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci. 2017, 42, 339. [CrossRef]
46. Ma, Z.; Liu, S. A review of 3D reconstruction techniques in civil engineering and their applications. Adv. Eng. Inform. 2018, 37, 163-174. [CrossRef]
47. Xie, Y.; Tian, J.; Zhu, X.X. Linking Points with Labels in 3D: A Review of Point Cloud Semantic Segmentation. IEEE Geosci. Remote Sens. Mag. 2020, 8, 38-59. [CrossRef]
48. Xu, Y.; Tuttas, S.; Hoegner, L.; Stilla, U. Reconstruction of scaffolds from a photogrammetric point cloud of construction sites using a novel 3D local feature descriptor. Autom. Constr. 2018, 85, 76-95. [CrossRef]
49. Krijnen, T.; Beetz, J. An IFC schema extension and binary serialization format to efficiently integrate point cloud data into building models. Adv. Eng. Inform. 2017, 33, 473-490. [CrossRef]
50. Li, L.; Yang, F.; Zhu, H.; Li, D.; Li, Y.; Tang, L. An improved RANSAC for 3D point cloud plane segmentation based on normal distribution transformation cells. Remote Sens. 2017, 9, 433. [CrossRef]
51. Ramík, D.M.; Sabourin, C.; Moreno, R.; Madani, K. A machine learning based intelligent vision system for autonomous object detection and recognition. Appl. Intell. 2014, 40, 358-375. [CrossRef]
52. Wang, Z.; Liu, H.; Qian, Y.; Xu, T. Real-time plane segmentation and obstacle detection of 3D point clouds for indoor scenes. In Proceedings of the European Conference on Computer Vision, Florence, Italy, 7-13 October 2012; Springer: Berlin/Heidelberg, Germany, 2012; pp. 22-31.
53. Vo, A.V.; Truong-Hong, L.; Laefer, D.F.; Bertolotto, M. Octree-based region growing for point cloud segmentation. ISPRS J. Photogramm. Remote Sens. 2015, 104, 88-100. [CrossRef]
54. Boerner, R.; Hoegner, L.; Stilla, U. Voxel Based Segmentation of Large Airborne Topobathymetric LIDAR Data. In International Archives of the Photogrammetry, Remote Sensing \& Spatial Information Sciences 42, Proceedings of the ISPRS Hannover Workshop: HRIGI 17-CMRT 17-ISA 17-EuroCOW 17, Hannover, Germany, 6-9 June 2017; ISPRS: Hannover, Germany, 2017.
55. Li, M. A Super Voxel-Based Riemannian Graph for Multi Scale Segmentation of LIDAR Point Clouds. In ISPRS Annals of Photogrammetry, Remote Sensing \& Spatial Information Sciences, 4, Proceedings of the ISPRS TC III Mid-Term Symposium "Developments, Technologies and Applications in Remote Sensing", Beijing, China, 7-10 May 2018; ISPRS: Hannover, Germany, 2018.
56. Huang, M.; Wei, P.; Liu, X. An Efficient Encoding Voxel-Based Segmentation (EVBS) Algorithm Based on Fast Adjacent Voxel Search for Point Cloud Plane Segmentation. Remote Sens. 2019, 11, 2727. [CrossRef]
57. Besl, P.J.; Jain, R.C. Segmentation through variable-order surface fitting. IEEE Trans. Pattern Anal. Mach. Intell. 1988, 10, 167-192. [CrossRef]
58. Zhan, Q.; Liang, Y.; Xiao, Y. Color-based segmentation of point clouds. Laser Scanning 2009, 38, 155-161.
59. Nurunnabi, A.; Belton, D.; West, G. Robust segmentation for large volumes of laser scanning three-dimensional point cloud data. IEEE Trans. Geosci. Remote Sens. 2016, 54, 4790-4805. [CrossRef]
60. Khaloo, A.; Lattanzi, D. Robust normal estimation and region growing segmentation of infrastructure 3D point cloud models. Adv. Eng. Inform. 2017, 34, 1-16. [CrossRef]
61. Wang, L.; Xu, Y.; Li, Y. Aerial LIDAR point cloud voxelization with its 3D ground filtering application. Photogramm. Eng. Remote Sens. 2017, 83, 95-107. [CrossRef]
62. Liu, L.; Xiao, J.; Wang, Y. Major orientation estimation-based rock surface extraction for 3d rock-mass point clouds. Remote Sens. 2019, 11, 635. [CrossRef]
63. Awwad, T.M.; Zhu, Q.; Du, Z.; Zhang, Y. An improved segmentation approach for planar surfaces from unstructured 3D point clouds. Photogramm. Rec. 2010, 25, 5-23. [CrossRef]
64. Cover, T.; Hart, P. Nearest neighbor pattern classification. IEEE Trans. Inf. Theory 1967, 13, 21-27. [CrossRef]
