SPACE SUBDIVISION IN INDOOR MOBILE LASER SCANNING POINT CLOUDS BASED ON SCANLINE ANALYSIS

YI ZHENG
February, 2018

SUPERVISORS:
Dr.ing, M.S, Peter
Dr.ir, S.J, Oude Elberink
Advisor: S., Nikoohemat MSc

SPACE SUBDIVISION IN INDOOR MOBILE LASER SCANNING POINT CLOUDS BASED ON SCANLINE ANALYSIS

YI ZHENG
Enschede, The Netherlands, February, 2018

Thesis submitted to the Faculty of Geo-Information Science and Earth Observation of the University of Twente in partial fulfilment of the requirements for the degree of Master of Science in Geo-information Science and Earth Observation.
Specialization: Geoinformatics

SUPERVISORS:
Dr.ing, M.S, Peter
Dr.ir, S.J, Oude Elberink
Advisor: S., Nikoohemat MSc

THESIS ASSESSMENT BOARD:
Prof.dr.ir, M.G, Vosselman (Chair)
Dr. M.N. Koeva (External Examiner, University of Twente, ITC-PGM)

DISCLAIMER

This document describes work undertaken as part of a programme of study at the Faculty of Geo-Information Science and Earth Observation of the University of Twente. All views and opinions expressed therein remain the sole responsibility of the author, and do not necessarily represent those of the Faculty.

ABSTRACT

Indoor scene shows board application in our daily life, such as navigation, mapping, disaster management etc. In order to meet the demands of these applications, the Indoor Mobile Laser Scanner (IMLS) is widely utilised for acquiring indoor information. This is due to IMLS system not only obtain point clouds at a sufficient level of detail and accuracy within a short period but also provides a continuous trajectory of device locations. To have a better understanding of the indoor scene, several objects like doors, windows and space subdivision are the object of this research, which provides important architectural structure for indoor environments.

This thesis proposes a workflow to subdivide space based on opening detection results. Up to now, many opening detection methods have been demonstrated based on trajectory and modelling which leads complicating operations, high computational load and low processing speed because they need analysis numerous points in the dataset. In this research, I demonstrated a novel opening detection method which can efficiently detects both doors and windows by analysing scanlines. This method could be even used in real-time processing.

After that, a two-step space subdivision method based on extracted openings is described. Firstly, the trajectory points are subdivided into different spaces. Secondly, the corresponding point clouds that are related to the trajectory points within same space are subdivided based on the detected doors. In the end, doors, windows and space subdivision results were saved as point cloud label, which will be used for further investigations.

The workflow has been tested on the real dataset that collected by ZEB-REVO from GeoSLAM Company. Finally, the experimental results validated the completeness and robustness of the proposed methods under different indoor environment and different scanning paths.

Keywords

Door detection, Space subdivision, Trajectory, Indoor point clouds

ACKNOWLEDGEMENTS

I would like to thank all those people who made this thesis possible and an unforgettable experience for me.

Special mention goes to my enthusiastic first supervisor, Dr Michael. My MSc has been an amazing experience and I thank Michael wholeheartedly for his tremendous academic support and warm encouragement. I would also like to thank my second supervisor Dr Sander and my advisor Shayan for the useful guidance and advice.

Each of the ITC staff and students has provided me extensive personal and professional guidance and taught me a great deal about both scientific research and life in general. My sincere thanks to Ahmed and chaoge for supporting and helping in the research. I should also thank Brook, for all him love and support.

Finally, I must express my very profound gratitude to my mum Rihong and dad Tingjun through the process of researching and writing this thesis. This accomplishment would not have been possible without them. Thank you.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

1.1. Motivation and Problem Statement

The Environmental Protection Agency has a survey of national human activity and it indicated that the average American spends 87% of their life in indoors (Klepeis et al., 2001). It suggests the great potentials and broad applications of indoor scene in daily life. For instance, crime scene investigation (Buck U et al., 2013), cultural heritage (Xu et al., 2014), indoor navigation (Peter et al., 2012) and mapping (Khoshelham & Elberink, 2012).

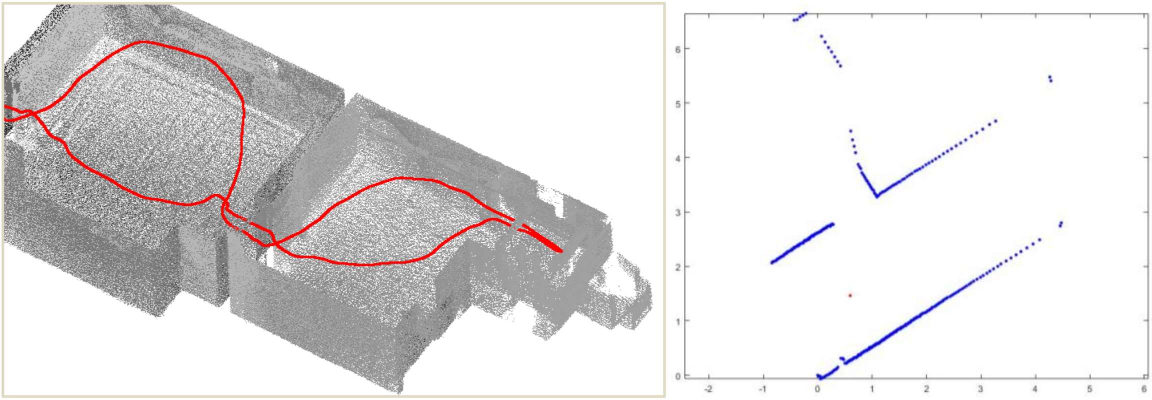
In order to meet the demands of these applications, laser scanners and RGB-D sensors have been widely utilized to acquire 3D point clouds in indoor environment. This is because they can acquire accurate 3D points within a short period. The RGB-D cameras are inexpensive and lightweight sensors that provide both color and depth information (Dos Santos et al., 2016), but the quality of acquired data is constrained by the illumination condition (Kerl et al., 2015) while laser scanners can obtain three-dimensional point cloud data directly in a fast and detailed way (Sahin, 2015). The main types for laser data collection are Mobile Laser Scanner (MLS) and Terrestrial Laser Scanner (TLS). Unlike TLS, which needs tie points and a series of manual works, MLS can obtain sufficient level of detail and accuracy within a short period. Hence, we use MLS to study indoor scenes in this research.

Because of the large and complex dataset, the lack of texture information as well as the noise and occlusions the procedure of classification is a primary challenging task (Hu & Ye, 2013). Few classification methods have been proposed to identify indoor scene by applying deep learning (Qi et al., 2016), boosting (Rottmann et al., 2005) or search-classify (Nan et al., 2012). However, these methods are based on the whole dataset, which needs complicating operations and leads to high computational load and low processing speed.

Hebel & Stilla (2008) and Hu & Ye (2013) showcase that the processing based on a scanline can be executed comparatively fast and uncomplicated compared to 3D analysis. During scanner captures the scanline each time, we could classify it directly. Consequently, the methods would be applicable in online data preparation for subsequent analysis or real-time classification. Nevertheless, these approaches operate to detect objects in airborne laser scanner data. In a word, analyzing scan line could be an efficient, real-time algorithm, even can be used in online processing.

One main drawback for the single scanline (see Figure 1-1 (b)) is that it provides limited information to recognize indoor scenes for occlusion, because it is represented by segments composed of a set of points. To overcome this drawback, I try to join analysis of trajectory and neighboring scanlines to get fine details of the indoor objects. Up to now, several algorithms have been proposed based on trajectory for the detection of indoor object or different places (Díaz-Vilariño et al., 2017; Koppula et al., 2011; Rottmann et al., 2005). However, these methods just look at the shape of trajectory which has limited capability to discriminate object that does not interact with the trajectory, such as the doors which are passed by the operator.

Thus, the analysis by joining trajectory and point cloud data is presented in this research. Figure 1-1 (a) shows a part of indoor point cloud and the corresponding trajectory of the scanning system. Each scanline connects with a point in trajectory by timestamp which could help to find their neighbor scanlines. The diacritical objects can be extracted by analyzing local geometric regularity in single scanline and their neighbors. This way, I can identify opening like walls and windows. Even the point cloud can be partitioned into different spaces based on the extracted information and trajectory.



(a) (b)

Figure 1-1: The example of point clouds (a) (grey: point cloud and red: trajectory) and 2D scanline (b) (blue: points and red: scanner location) from Zeb-REVO.

1.2. Research Identification

The goal of this research is to detect openings and subdivide space based on scanline analysis, and partition space by joint analysis of trajectory, point cloud and extracted door. It can be used in many specific applications. Typical examples are simultaneous update of indoor scene in mobile robots and Simultaneous Localization and Mapping (SLAM).

1.2.1. Research Objectives

The overall objectives of this research are to extract openings (doors and windows) and partition space. To meet the general objects, I subdivided it into the following sub-objectives:

1. Design and implement the appropriate opening detection methods that are capable of distinguishing door and window candidate in single scanline.
2. Use neighboring scan lines for analysis to extract doors and windows.
3. Space partitioning by joint analysis of trajectory and doorframe.
4. Analyzing extracted results.

1.2.2. Research Questions

The following research questions should be answered in order to achieve the above objects.

Sub-objective 1:

- Which objects can be extracted and detected in single scanline?
- Which features can be utilized in detection processing?
- What is the distribution of the noise and how can it be removed?

Sub-objective 2:

- Will the adjacent scan line beneficial to recognize indoor objects? If it is, how many?
- How to use adjacent scan line to improve detection results?

Sub-objective 3:

- Are the trajectory and extracted objects useful to special partitioning?

Sub-objective 4:

- How to evaluate the results?

1.3. Innovation

The innovation of this research lies in design, implement a novel door and window detection method based on single scan line and their neighbours. The method based on scanlines will reduce the dimensionality of algorithm and processing time. Consequently, this method could be applied to online processing, SLAM and mobile robots.

1.4. Thesis Structure

There are five chapters in this thesis. This chapter is gives a general concept regarding motivation, research objects, questions and innovation. Some relevant works are presented in Chapter 2. Chapter 3 describes the designed methodology and parameter analysis. The fourth chapter shows the performance of a set of steps in two datasets and discusses the results. The final chapter contains conclusion and recommendations.

2. LITERATURE REVIEW

This chapter reviews the existing classification and segmentation methods. I start with indoor 3D acquisition sensors followed by a brief description. Two different scanners are described and compared in this section. Several popular indoor laser scanners are also provided. After that, some state-of-the-art methodologies of point cloud will be presented in the two main aspects: segmentation and classification. Moreover, a short summary of literature review is given in the end.

2.1. Indoor 3D Acquisition Sensors

There are two basic scanners for optically measuring which have been widely utilized to acquire indoor 3D information: RGB-D cameras and laser scanner. This is because they can accurately and robustly acquire 3D points within a short period.

2.1.1. RGB-D Camera

The RGB-D camera is a specific type of depth sensing devices that work in association with RGB camera, can augment the conventional image with depth information in a per-pixel basis (Agapito et al., 2015). RGB-D sensors, such as Microsoft's Kinect as in Figure 2-1 (a), use RGB camera either combined with stereo sensing or time-of-flight laser sensing to generate depth estimates that can be associated with RGB pixels (Henry et al., 2010). Hence, the point cloud that is constructed the depth image can have additional colour feature.

As a light-weight and low-cost scanner, RGB-D camera becomes an attractive alternative to laser scanners in application areas such as object detection, cultural heritage conservation and indoor mapping. Pham et al. (2015) proposed a method that uses higher-order and hierarchical CRF model for semantic labelling in 3D indoor scene. Quintana et al. (2016) demonstrated a novel approach that based on detection of rectangular point cloud data gaps to recognize open and closed doors by using the information provided by both RGB camera and 3D laser scanner. A graphical model that captures various features and contextual relations for scene understanding was presented by Koppula et al. (2011).

However, the limitation of RGB-D scanners, as a sort of active sensing devices, have a very limited measurement range and colour features rely on captured images. Ishikawa et al. (2016), Susperregi et al. (2013) and Wen et al. (2014) used a multiple sensor fusion approach to cope with this limitation. And the result of these research shows that using complementary sensors and fusing their results could obtain high density and accuracy 3D data.

2.1.2. Laser Scanner

The term laser scanner covers time-of-flight (TOF) and phase-based laser scanners. TOF scanner measure the light traveling time from a source to a reflective target and back to the source (Vosselman & Maas, 2010). Compare to the RGB-D scanner, Laser scanner can obtain dense and accurate three-dimensional point cloud directly, and it greatly improves the speed of data collection. Currently, the main types for laser data collection are divided mainly into two techniques: Terrestrial Laser Scanning (TLS) and Mobile Laser Scanning (MLS).

Terrestrial Laser Scanners, like Leica C10 in (b) perform from a static position during the data acquisition. It is feasible for detailed small-area surveys having a typical radius for a high-density scan of less than a few tens of meters (Holopainen et al., 2013). Hence terrestrial laser scanners are being used more frequently for the reconstruction of building models, cultural heritage and civil engineering. But it is time-consuming for data acquisition of TLS due to the need for tie points and a series of manual works. Several authors have proposed different strategies for point cloud classification by using terrestrial laser scanning data. For example, Luo & Sohn (2013) used a conditional random field for line-based classification that labels TLS point clouds into vertical object, ground, tree and low objects.

Mobile laser scanning is an emerging technology capable of obtaining sufficient level of detail and accuracy within a short period. MLS can obtain sufficient level of detail and accuracy within a short period. Many of the applications involve mobile mapping as well as indoor modelling are use mobile laser scanner in data acquisition (Kukko et al., 2012). In general, the mobile laser scanner has to be supplemented by GPS unit and Inertia Measurement Device (IMU) to acquire position information.

Indoor mobile laser scanners are usually mounted on backpacks, push-car or held by hand that can move with operator. As illustrated in Figure 2.1 (c), the ZEB-REVO is a light-weight handheld laser scanner which consists of a 2D laser range scanner coupled to an IMU mounted on a motor drive. During the measurement, the rotating scanner head scans the surrounding with a high precision laser which captures more than 40,000 surface points per second. In the end, this system saves dataset as raw 2D points and inertial data. For generate 3D point clouds, it combines the 2D data with the IMU information by using a simultaneous localization and mapping (SLAM) algorithm.

(a) Microsoft Kinect (b) Leica C10 (c) ZEB-REVO

Figure 2-1: Indoor laser scanner

2.2. Segmentation

Segmentation is the process of detecting subsets in point clouds that are characterized by one or more characteristics in common. Point cloud segmentation plays an important role in object recognition and detection, since it can support classification as well as provide further feature extraction (Aijazi et al., 2016). Nowadays, some techniques for the segmentation have been developed for the extraction of 3d planes. There are also several methodologies working on 2D laser scanning data.

2.2.1. Segmentation into 3D Planes

Some of the segmentation methods based on geometrical and radiometric characteristics, such as point position, locally estimated surface normal, residuals of best fitting surface procedures, points reflectance, etc. (Grilli et al., 2017). Shui et al. (2016) presented a coarse-to-fine planar shape segmentation method that based on region growing and RANSAC to segment planar shapes from complicated indoor scenes containing. A robust randomized shape detection algorithm based on random sampling to detects planes, spheres, cylinders, cones and tori was demonstrated by Schnabel et al. (2007). In another study, Zancajo-Blazquez et al. (2015) proposed a methodology for the segmentation based on Principal Component Analysis (PCA) with the covariance method and the computation of the eigenvalues and eigenvectors. These methods require a fine-tuning of different parameters depending upon the nature of data and applications (Grilli et al., 2017).

Deep learning and machine learning have emerged as another popular method for semantic segmentation. Qi et al. (2016) design an efficient and effective neural network (PointNet) for part segmentation and semantic segmentation. Tschannen et al. (2016) proposed a highly structured CNN architecture that combines a tree-like CNN-based feature extractor, a random layer realizing a radial basis function (RBF) kernel approximation for semantic segmentation. The weakness is these methods need large labeled data sets and requires parameter optimization.

2.2.2. Segmentation of 2D Laser Scans

There are three different segmentation algorithms for 2D laser scanning data: Successive Edge Following (SEF), Line Tracking (LT) as well as Iterative End Point Fit (IEPF) (Siadat et al., 1997; Nguyen et al., 2005; Peter et al., 2017). There are several methods are proposed to improve these segmentation algorithms. A kind of segmentation method based on testing a range of residuals was used on getting segment for single scan line which can produce accurate linear segments and preventing tilted segments (Peter et al., 2017). Hebel & Stilla (2008) use RANSAC method to fit line segments of 2D points in a scan line. According to their study, the number of inliers and the average distance of all inliers to the line are used to evaluate the quality of the fitted straight line.

2.3. Classification

The classification has been a core topic of laser scanning as well as computer vision for many years. The goal of the classification is to define and assign points to specific classes according to different criteria (Grilli et al., 2017). Many researchers have tried to use various methods to classify point cloud.

A fast scene classification algorithm to identify and categorize indoor environments based on onboard sensors was proposed by Shi et al. (2011). The strategy was based on a support vector machines classifier as well as an efficient feature selection algorithm.

A search-classify approach is another method used for indoor scene understanding and point cloud classification(Nan et al., 2012). The key idea of this technique is to interleave the computations of segmentation and classification of the scene into meaningful parts of object. The algorithm can fit templates to objects, and is robust in noise and clutter. However, the drawback of this method is that the template fitting uses the upward direction with respect to the floor, which means objects not obey this assumption will yield wrong classification result.

In another study, Mura et al. (2014) proposed a method that combines occlusion-aware process and diffusion process for automatic reconstruction of indoor environment. It was concluded that this method is robust against clutter and occlusions, but this method did not detect opening area like window and door, neither perform well to recognize slanted walls, ceiling of different heights and staircases.

Another study by Mattausch et al. (2014) used a patch similarity measure based on shape descriptors and spatial configurations of neighbouring patches to detect repeated indoor objects, like chair, cupboard, lamps and more. The method did not need any training and labelled dataset, and robustly filters out noise as well as effectively reduce computational cost and memory requirement, but the weakness is that it cannot represent small object that have many planar regions.

In comparison, the processing based on scan line will faster and simpler. Hu & Ye (2013) proposed a Douglas-Peucker algorithm for segment the scan line into segment objects based on height variation and set a simple rule-based classification to distinguish building and non-building in outdoor environment. The authors claim the method is still needed 2D or 3D neighbourhood to meet higher detection quality. But the rules of classification it too simple to appropriate for analysis indoor scene which has complex, occlude environment, since it just considered the length of segments and the distance from points to digital elevation model.

In another study, use of real-time capable filter operation that based on random sample consensus to distinguish clutter and men-made objects was demonstrated by Hebel & Stilla (2008). According to their study, a line-growing algorithm was used to form a connect surface in consecutive scan lines. It was concluded that 3D classification methods are time consuming and high algorithm complexity.

As for door detection, Díaz-Vilariño et al. (2017) presented a method to detect local minimum of vertical distances based on a combination of trajectory. And Ochmann et al. (2014) set the rules to detect opening doors based on analysis the points on the overlap area. A method that use the point behind the detected façades planes is described by Tuttas & Stilla (2009). Another door detection method based on the detection of rectangular point cloud data gaps in the wall planes is proposed by Quintana et al. (2016).

3. RESEARCH METHODOLOGY

This chapter is organized as follows: Section 3.1 gives an introduction to the overall methodology to meet the research objectives. Following this, each step is described in detail. A description of the data acquisition is given in Section 3.2. Section 3.3 introduces the process of data pre-processing. The method of segmentation is presented in Section 3.4. Section 3.5 is devoted to illustrating the detail of extracted features. A final step is classification, how to classify point cloud, join analysis of neighbouring scanlines and more details which will be illustrated in Section 3.6.

3.1. Introduction

The general framework of this research is shown in Figure 3-1. The dataset which contains the point clouds and the trajectory was acquired by the 2D laser scanners. In order to determine the effectiveness of real-time processing, pre-processing is used to restore the scanlines. After that, segmentation and feature extraction are processed on each scanline separately. Some points, such as doorframe points, are differentiated and extracted to assign semantic labels by analysing the features of segments. To acquire reliable classification results, the neighbour scanlines will join analysis after project the scanline to world coordinate system. In the end, doors, windows and space subdivision results will be saved as point clouds labelling, which will be used for further investigations.

It should be noted that the proposed method works on SLAM-based laser scanner, like ZEB-REVO used in this research, contains accurate and integrates scanlines and trajectory during data acquisition. This is because the scanlines' orientation and translation with respect to the world coordinate systems origin will not be identified if without SLAM algorithm.

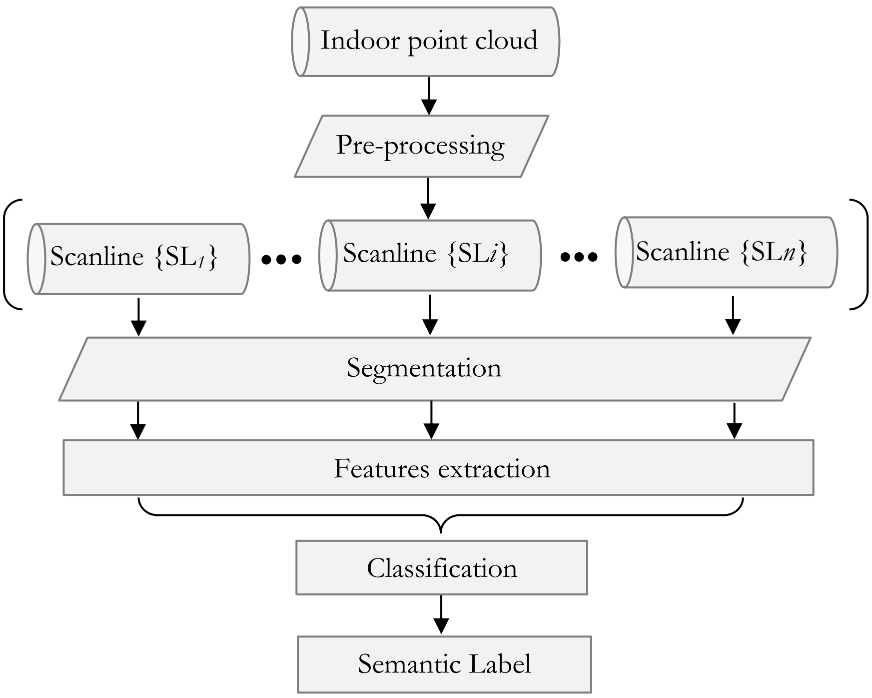


Figure 3-1: The workflow of classification methodology

3.2. Data Acquisition

The datasets were acquired by Zeb-REVO, a hand-held laser scanner, which has been described in Section 2.1.2. It is equipped with a laser range scanner, an IMU and a motor drive. The characteristics of Zeb-REVO laser device are shown in the Table 1. The laser in this scanning system is a 2D time-of-light laser with 270 degrees of view, 905 nm laser wavelength, up to 30 m scanning range and ±30mm scan range noise.

Table 1:Technical characteristics of the Zeb-REVO according to the manufacturer datasheet.

Point per scan line	432(0.625°interval)
Field of view	270°*360°
Scan rate	100 lines/s 43200 points/s

During the data acquisition process, the rotating 2D scanner head rotates around the roll axis of the system and scans the indoor environment with a high precision scanline which captures around 100 scanlines per second. The quaternion is converted to Euler angles (roll, pitch and yaw) to represent how the pose of scanner changed during data acquisition, which is because they are conceptually easier to understand. Each value of Euler angles represents the rotation in degrees around one of the x, y and z axis in world coordinate system.

The scanner orientation changing over time can be seen in Figure 3-2. The roll angle changed regularly as the scanner head rotate steadily. The pitch angle always keeps around 0°, this is because operator hold the instrument perpendicular to the floor. The changing yaw angle indicates the changes in horizontal orientation of the operator.

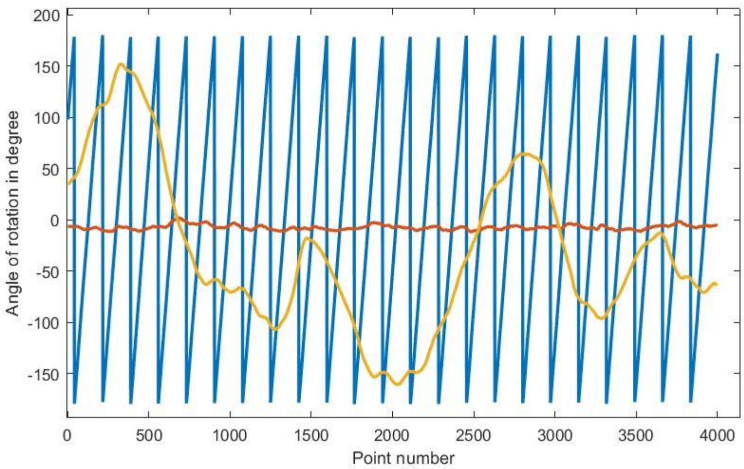
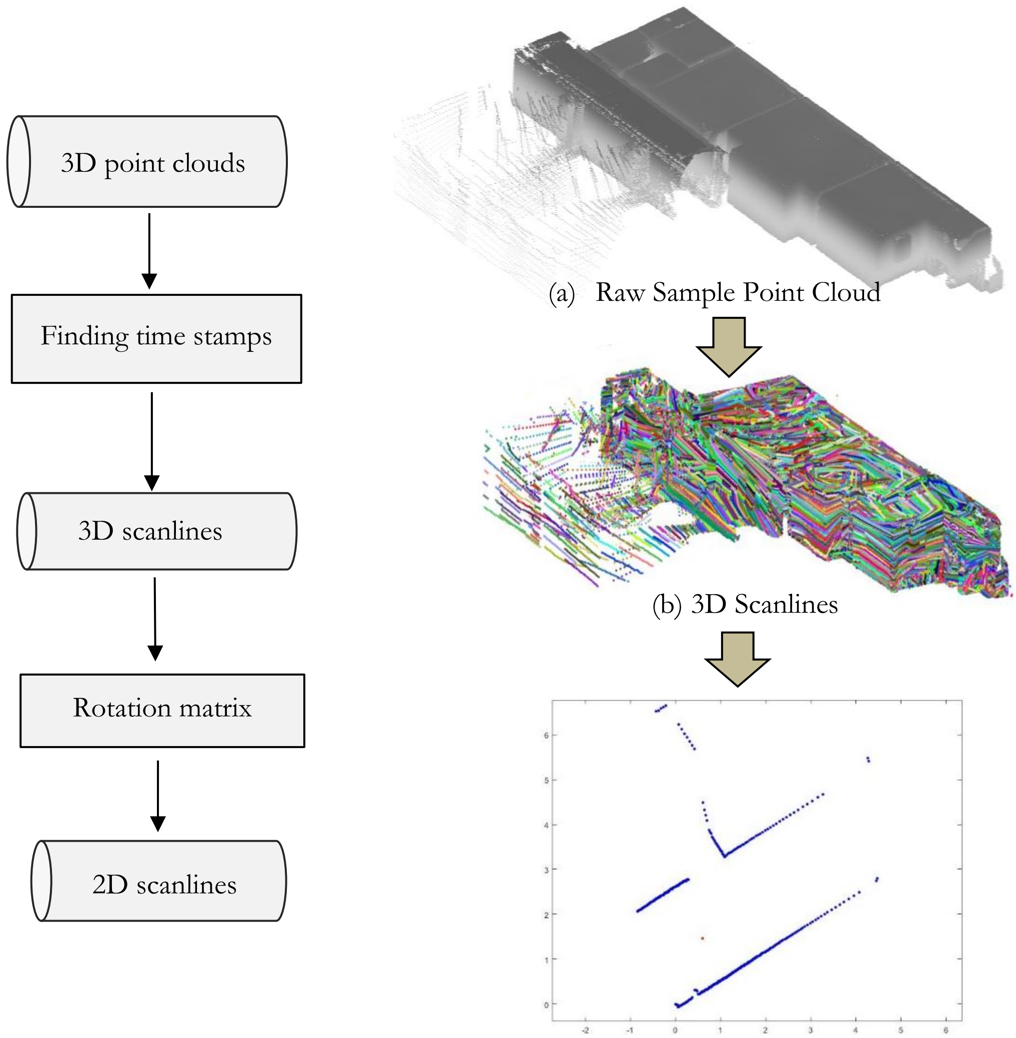


Figure 3-2: An example of Scanner's attitude (blue: roll angle; red: pitch angle; yellow: yaw angle).

3.3. Data Pre-processing

Prior to segmentation, the time stamp is needed to split the whole point clouds into scanlines. The main intention of this step is restoring the data as it actually was like during data acquisition. A general diagram presenting the pre-processing process is shown in Figure 3-3.



(c) Example of a 2D horizontal Scanline

Figure 3-3: Flow-chart of pre-processing illustrated with a sample point cloud. (a) Raw sample point clouds coded with height values (b) 3D view of whole scanlines (different colours label different scanlines) (c) Projection result of a horizontal scanline.

3.3.1. Method Description

As described in Section 3.1, the input data of this research is 3D point clouds. The main idea is use time difference between neighbour points to find timestamp. Figure 3-4 shows an example of single scanline. In principle, the constant rotating scanner will send out laser beam each time when it rotated 0.625° in same scanline, and it will stop to send out laser beam until rotate 90° after this scanline (as shown in Table 1). Consequently, the time difference value between points within the same scanline is smaller.

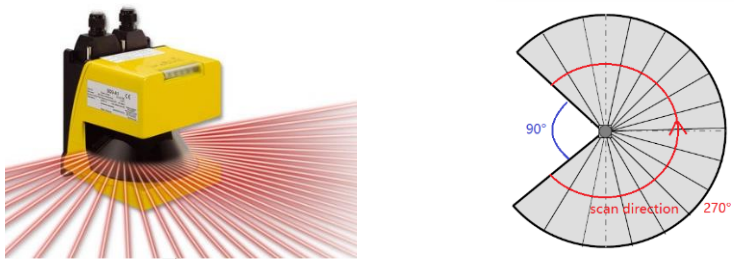


Figure 3-4: Example of scanning angle in 3D (left) and 2D (right).

However, in practice not all scanlines have exactly 432 points, since surfaces that are more than 30m away from the scanner will not result in a valid signal. This situation may let bigger time difference within the same scanline. Considering this fact, a threshold th_{time} can be used to defined if the time difference of two neighbour points is large enough to separate point cloud into two scanlines. Once the time difference is higher than the threshold we conclude that the times stamp is between their corresponding times.

Once the 3D scanlines are extracted, they are projected into 2D space, which are the input of segmentation. Projection is a procedure that restore the data as it actually was like during data acquisition. This step in other words can be seen as the transformation from world coordinate system to scanner coordinate system. Suppose the position of the scanner in world coordinates is P_0, the scanline points in world coordinate is Pw [Xw, Yw, Zw] and it corresponding position in scanner coordinate system is Psl [Xsl, Ysl, Zsl]. The coordinate value of Psl can be calculated by Equation 3.1:

$$\begin{bmatrix} Xsl \\ Ysl \\ Zsl \\ 1 \end{bmatrix} = \begin{bmatrix} R & -RP_0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} Xw \\ Yw \\ Zw \\ 1 \end{bmatrix} \tag{3.1}$$

Where R is the rotation matrix that converted from quaternion. The list of parameters used during pre-processing is shown in Table 2.

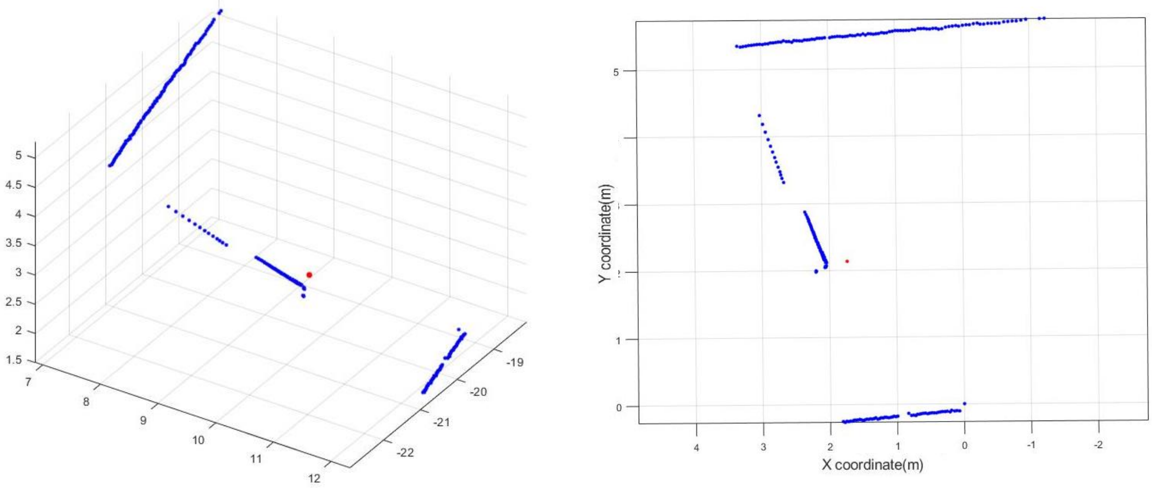


Figure 3-5: The example scanline before (left) and after (right) projection.

Table 2: Parameters used in the pre-processing. (Default parameters shows in white rows)

Parameter Name	Description
th_{time}	The threshold of time difference to find out different scanlines

3.3.2. Parameter Analysis

The parameter th_{time} depends on the scanning system. Time difference, in general, is related to the scanning frequency. The Figure 3-6 shows the time difference between points in a 3-d dataset with 5000 points. The vertical (y) axis represents the time different, while the horizontal (x) axis represents the sequence of inputted point 1,2 …, 5000. It supports the fact that the time difference between scanlines will be larger than neighbour points within same scanline obviously. After analysing the plot of time difference, we set th_{time} as $2.25*10^{-3}$.

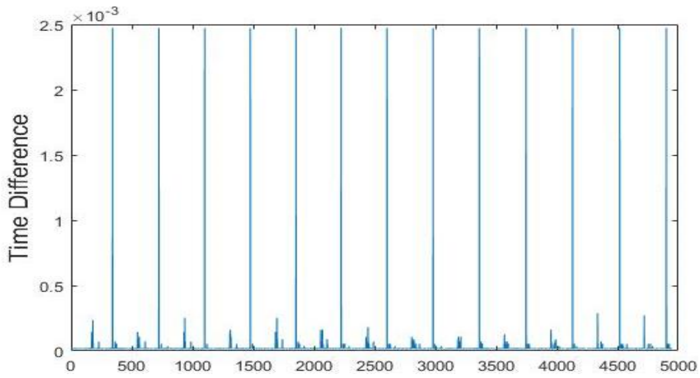


Figure 3-6: The time difference of neighbour points

To improve and check the quality of the scanline, the eigenvalues (λ1, λ2, λ3) can be applied to evaluate the result of pre-processing. In point clouds, the eigenvalues represent the variance of the coordinates of all points along the eigenvector. Eigen-features that derived from eigenvalues are commonly used to describe the local geometric characteristics as well as present check and prove whether the local geometry is planar, spherical and so forth (Lin et al., 2014). The basic assumption of the scanline is: if all points belong to the same scanline, they should lie on the same plane. In order to describe the geometric characteristics and indicate whether the geometry of an extracted single scanline is planar, eigenvalue is applied to evaluate the result of pre-processing. If certain scanlines are considered as not on the same plane, which means there are at least two scanlines in this extracted scanline point set. Therefore, a smaller threshold will be used to split it, and use eigen-features to analyse new the result again.

3.4. Segmentation

In indoor environment, typic objects, like ceiling, floor as well as wall, will present as straight-line segments within the scanline. Moreover, segments can carry more information than analysing the point distribution in a local neighbourhood, and some segment features, such as the line vector, are stable and useful for classification (Vosselman et al., 2017).

3.4.1. Method Description

In this research, a line segmentation method was used to split single scanline into linear segments. The result of pre-processed 2D scanline, was used as input (as shown in Figure 8 (f)). This segmentation approach first used a number of points to fit a line and accepted it as a candidate if the mean values of a range of residuals is lower than a metric threshold. The following step combined the results of forward and backward processing to produce accurate linear segments and prevent tilted segments. The points will be labelled as belonging to different segments after the segmentation process. The detail of this algorithm was described in (Peter et al., 2017). Table 3 contains a list of the parameters used in the segmentation process with a brief description.

Table 3: Parameters used in the line segmentation process. (User-specified parameters shows in grey rows)

Parameter Name	Description
minLen	Minimum number of points in a segment considered to form a reliable segment.
percentage	The percentage of points (which are part of the segment) to use for testing
N	The number of points being used for initialization of a segment
minTH	Minimum number of mean distances above thresholds
σ	A metric threshold

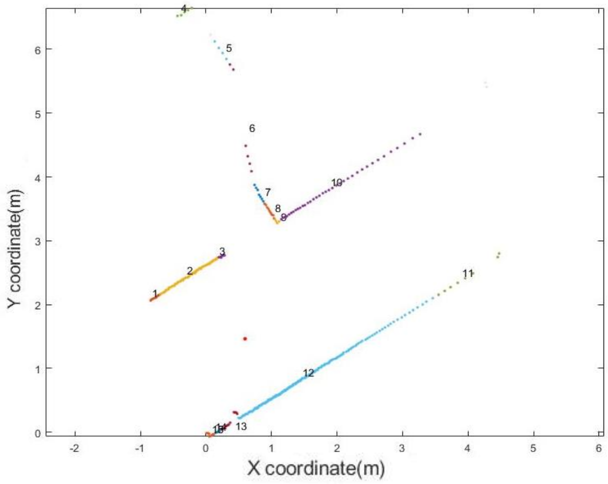
3.4.2. Parameter Analysis

As mentioned in the previous section, there are four main parameters which effect the result of segmentation. For analysis of the effect of different parameters and make a reliable visual comparison. Only varying one of the parameters where varied each time while keeping the others constant. Hence, it was repeated four times and the value shows in below.

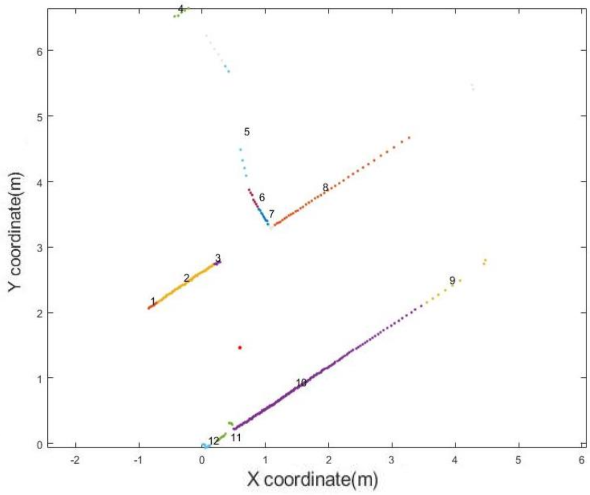
Table 4. Different segmentation parameter combinations

No.	*minLen*	*percentage*	*minTH*	σ
1	3	0.15	2	0.02
2	5	0.15	2	0.02
3	3	0.30	2	0.02
4	3	0.15	4	0.02
5	3	0.15	2	0.01

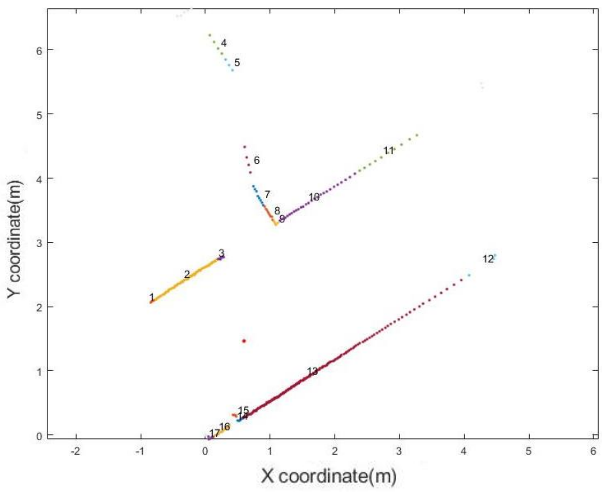
In the implementation, different result is plotted (as shown in Figure 3-7) corresponding to the parameter combination respectively. The input parameter *minLen* affect if there are some small or unreliable segments to be discarded. It also can be seen as a threshold to remove small segments and apparent anomaly points. Compared the result of Figure 3-7 (a) and Figure 3-7 (b), the segment 5 in Figure 3-7 (a) was discarded. Hence, small segments will be discarded when we set big *minLen*. Figure 3-7 (c) shows the influence of the *percentage*; bigger the value of *percentage*, easier the segment be over-segmented. Hence, the segment 5 and 12 in Figure 3-7 (a) where over-segmented when compare to the Figure 3-7 (c). For bigger *minTH*, smaller segments were discarded or under-segmented, like the segment 5 in Figure 3-7 (a) was discarded when compare with the Figure 3-7 (d). As for σ, I found small threshold might lead over-segmentation, like the segment 12 in Figure 3-7 (a) is over-segment into segment 13 and 14 in Figure 3-7 (e). It can be concluded that the first parameter combination not only can distinguish different segment but also identify short segment (like segment 4 in Figure 3-7 (a)). Note that, although it may generate over-segmentation result, it can extract most part of the segment as one segment.



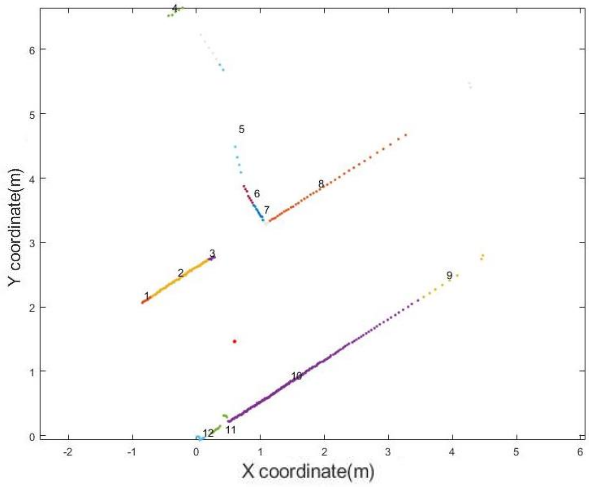
(a) Best result



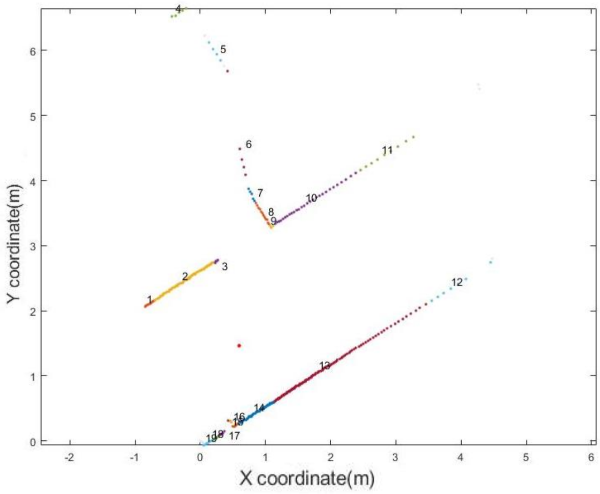
(b) Influence of *minLen*, *minLen* = 5



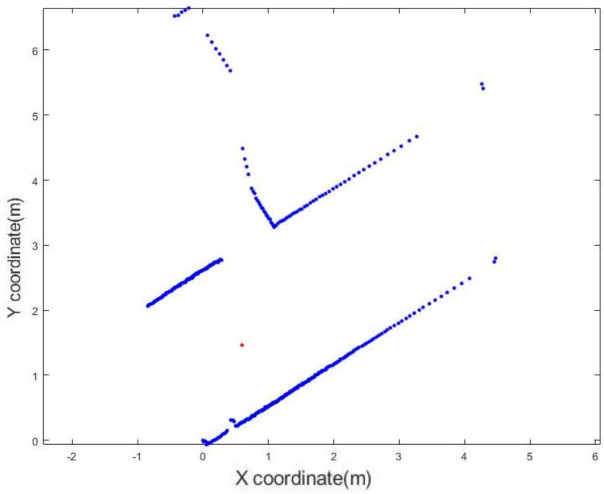
(c) Influence of *percentage*, *percentage*=0.30



(d) Influence of *minTH*, *minTH*=4



(e) Influence of σ, σ=0.01



(f) Raw example scanline

Figure 3-7: Segmentation results for different combinations of the input parameters (plot (a) - (e) related to each parameter combination in Table 4 respectively, different colours label different scanlines)

3.5. Feature Extraction

Generating features is an essential task because it can help us gain knowledge about the local environment around a segment. In this section, the features are achieved by analysing the geometric features and local contextual information, which could be used as input to build classifier in the next section.

Suppose we are given a number of segments 1, …, i that are extracted from one scanline. The segment features that are proposed in this section are separated into two types: segment features and features of a pair of segments. Segment features derived from analysis the point distribution in same segment. While another feature type describes the relation between a pair of segments.

For each segment i, the segment features that used in my research are summarized in the flowing:

1. Segment length Li:
The Euclidean distance between first point and last point of segment i
2. Segment size Si:
The number of points in the segment i
3. Normal vector and line vector of segment i:
To estimate normal vector and line vector, I use eigenvalues that generated by PCA (Principal component analysis) of segment.
4. Distance between scanner location and endpoints li:
The Euclidean distance between centroid and endpoint (first point or last point) of segment i (as shown in Figure 3-8).
5. The centroid of the segment i:
We calculate the average values of the coordinate of points in the segment i. It can be seen as the mean distance from scanner position to the segment.
6. Perpendicular distance from centroid to segment d_{0i}:
Let $ax + by + c = 0$ be the linear equation of the segment that passes through those points, where a and b are element of normal vector. The distance can be obtained by Equation 3.1:

$$d = \frac{|ax_0 + by_0 + c\ |}{\sqrt{a^2 + b^2}} \tag{3.1}$$

Where x_0 and y_0 are the scanner position.

Suppose that segment j is the neighbour segment of segment i. The features of a pair segments are showing below:

1. The scanning angle between segments θ:
The angle is intuitively illustrated in Figure 3-8. It is defined with the scanner position being the vertex, and the nearest point of a pair of segments on the legs.
2. Perpendicular and parallel:
We can produce topological relations between a pair of segments based on included angle of normal vector. The angle between a pair of segments was calculated using:

$$\theta = acos(\frac{\vec{a} \cdot \vec{b}}{|a| \cdot |b|}) \tag{3.2}$$

Where θ is the angle between segments, $\vec{a}$ and $\vec{b}$ are the normal vector of segments respectively. To find a segment parallel or perpendicular to another, we use a range variable of 4° considered in this case. Since it is rare to find perpendicular/parallel definitely.

3. Closest distance between neighbor segments $d_{closest}$:
The Euclidean distance between nearest point of segment *i* and *j*.

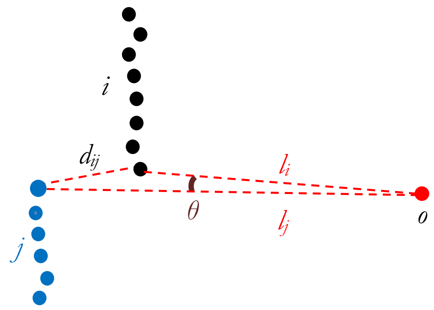


Figure 3-8: Example of neighbour segments

3.6. Classification

The main task of classification is to label $L \in \{1, \dots K\}$ for each point. In this research, we are interested in doors and windows and space subdivision information. Hence, how generated features can be used to extract semantic information is the main problem. Here, we integrate generated features and our knowledge of the indoor environment to detect objects based on scanline analysis.

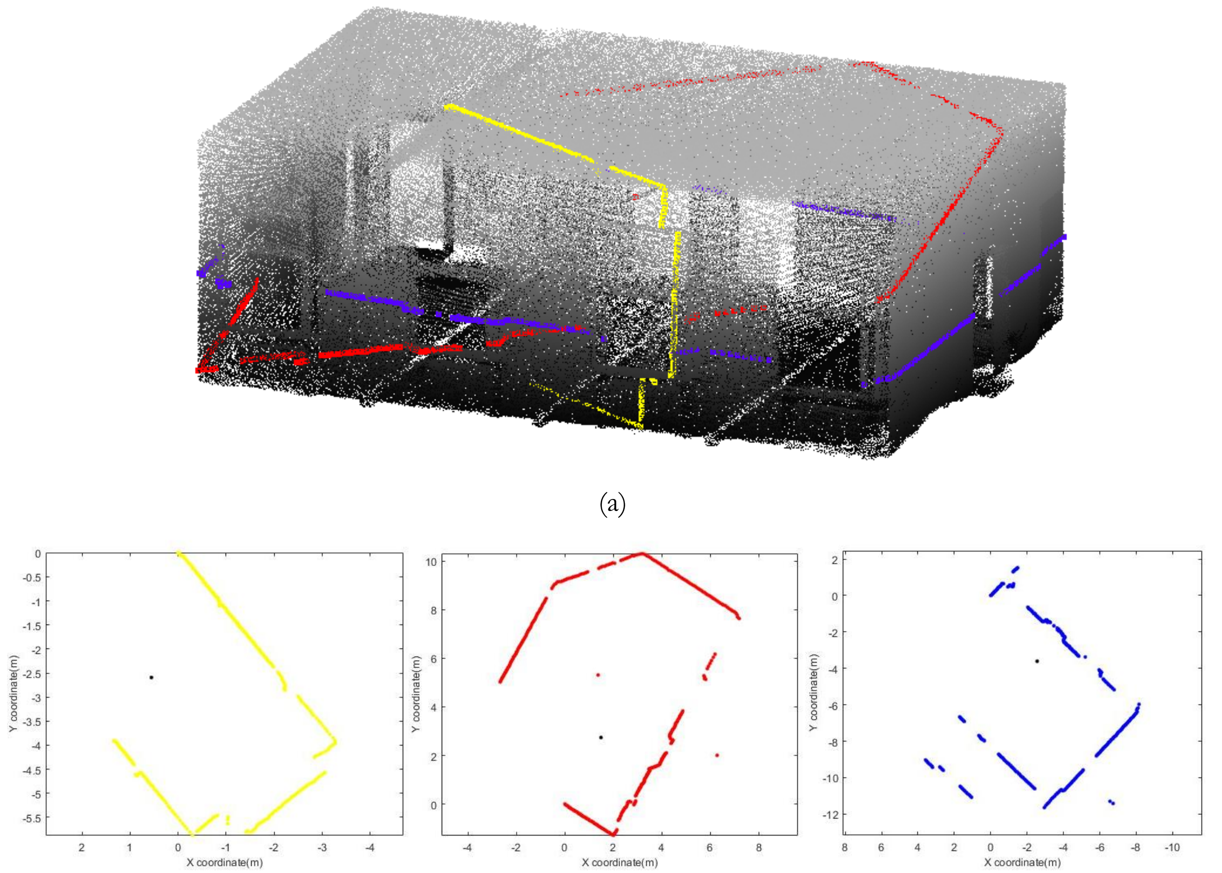
3.6.1. Ceiling/floor Distance Function

This proposed function is used to extract the distance from scanner to ceiling/floor (scanner height) which can be helpful for following classification. For example, distinguishing windows from opening detection due to the distance between window and floor.

The scanlines contain information about the surrounding environment. However, scanlines that are acquired in different roll angles always show different distributions and provide different information in indoor space (As shown in Figure 3-9). For example, horizontal scanline shows mainly information about the walls, and the vertical scanline shows the information about ceiling and floor normally. Unlike slant scanlines that may scan the wall beside of the operator, vertical scanlines that are acquired by ZEB-REVO always acquires the point in front of the operator. Hence, vertical scanline is a good choice to find the distance from scanner to ceiling or floor.

3.6.1.1. Method Description

In principle, the vertical scanlines' roll angle should be 90°/-90°. However, it is hard to find the exactly vertical scanlines. Therefore, I use an angle range *th_angle* to extract some scanlines that are nearly vertical. It should be noticed that these distances can only be acquired each time when the scanline is considered as vertical.



(a)

(b) (c) (d)

Figure 3-9: Example scanlines in different roll angles. (a) the overview of these different scanlines in 3D dataset. (b) roll angle=87.4098° (c) roll angle=177.9159° (d) roll angle=-162.6193° (black point is scanner position).

Figure 3-9 provides an example of nearly vertical scanline. In 3D space, it is easy to identify ceiling and floor segments based on the normal vector and z value of the segment. However, as for 2D scanline (Figure 3-9 (b)), it is hard to identify ceiling/floor segment since it lost both height information and top-down direction. The main idea to solve this problem is to project the direction vector of z-axis [0,0,1] to the 2d plane. Since if ceiling and floor planes are perpendicular with z-axis in 3D space, ceiling and floor segments will also perpendicular with projected vector V' in 2D space. Moreover, other features, such as segments' line vector, are also used in ceiling/floor detection. The process of this function is executed in the flowing steps:

1. Get the vector *V'* by project 3d vector [0,0,1] to the 2d plane.
2. Find the horizontal segment above or below the centroid based on vector V'.
3. Find the most reliable segment by analysis candidates.
4. Use the roll angle of scanline to find the accurate distance.

Figure 3-10 shows an example of vertical scanline. The vector *V'* was projected by projection matrix (as specified in Section 3.1.3). Second step is to generate ceiling/floor candidates. In this step, segments are assigned as candidates while the included angle between line vector and vector *V'* in the threshold *th_vertical*. Let vector *vs* link from centroid of scanline to the centroid of segment. Since vector V' always point to the ceiling, if the vector *vs* shows the same direction with V', which means this segment could be ceiling, and vice versa. In this example, the ceiling candidates are segment 8, 7 and 6. Meanwhile, the floor candidates are segment 1, 2 and 3. In order to find most reliable ceiling or floor segment, we considered segment length, direction distance from centroid o to the segment and the spatial relationship. For some segments with length (L_i) less than 1m, are judged not likely to be ceiling or floor. Although the segmentation method may generate over-segmentation result, the method will generally generate a long sub-segment and some short sub-segments rather than several mid-length segments (have discussed in Section 3.4.2). Therefore, this rule can help us remove some short segments, which might be light or pipeline in indoor environment. After short segments are removed, the direct distance d_{0i} are calculated as the candidate distance from centroid to ceiling/floor. There are several situations in extracted distances:

- More than one candidate distances from centroid to ceiling/floor:
  If the distance difference within a small threshold, let the mean of these distance be the final distance.
  If not, let line L_0 parallel to vector V' and passing through the centroid of the segment. Finding the closest segment S' to line L_0 and let d_{0i} of the segment S' as the final distance.
- Only one candidate distance: let d_{0i} be the final distance.

In the last step, we use roll angle to adjust the distance to ceiling and floor in 3d space. Let th denote the roll angle and the adjusted distance is:

$$d_{adjusted} = \cos(|th| - 90) * d^* \tag{3.3}$$

Where $d_{adjusted}$ is the adjusted distance, d^* is either of the dc and df.

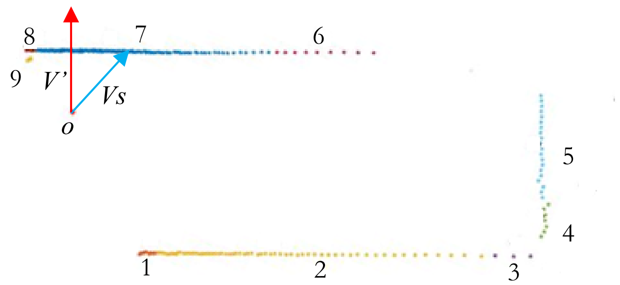


Figure 3-10: Example of a vertical scanline (different segment shown in a different colours; red point: scanner location; red line: projected direction vector of z-axis).

Table 5: Parameters used in the ceiling/floor distance function.

Parameter Name	Description
th_vertical	Angle range *th_angle* to extract nearly vertical scanline.

3.6.1.2. Parameter Analysis

The angle range (*th_vertical*) is used for vertical scanline extraction. If we set small value of *th_vertical*, it is possible miss the vertical scanline. In this scanning system, the roll angle difference between neighbour scanlines about 2°. Hence, we let *th_vertical =2°*, which means we are searching the scanline with roll angle in [88° 92°] or [-92° -88°].

3.6.2. Occlusion Function

This section proposed a method to detect the occlusion based on single scanline. Although occlusion relations were not used for object detection and spatial subdivision, it is an important information in indoor reconstruction, which can fill the architectural structures later. Figure 3-11 shows an example of the 3D occlusion. It can be observed that a part of the wall plane is occluded by door plane. But in the 2D scanline, this occlusion relation shows like Figure 3-11(right).

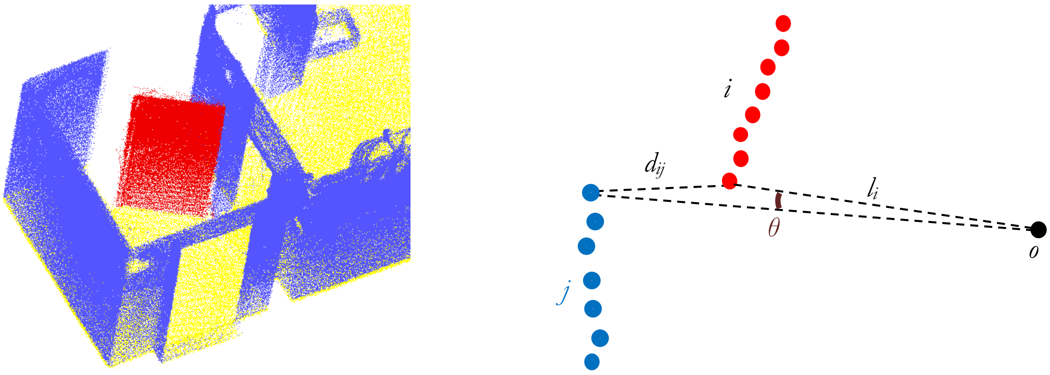


Figure 3-11: The example of occlusion in 3D plane (left, by manually extracted, red: door; blue: walls; yellow: floor) and 2D (right, different colour stands for different segments)

3.6.2.1. Method Description

Here, we only extract the candidate of occlusion relation between neighbouring segments, since it is difficult to determine whether there is occlusion relation between non-adjacent segments. Figure 3-11 (right) shows in red the segment i and in blue the segment j. Moreover, point o is the centroid of scanline. The pair of segments with occlusion relation should satisfy the following constraints:

- Angle θ between segment i and j should smaller than threshold th_{angle}.
- Closest distance $d_{closest}$ between segment i and j more than dynamic distance d_{dis}.

In the case of occlusion, the next ray (after the scanner moved by the angle increment) will return a point generally. In another word, the angle interval θ remains stable in the same scanline. If the angular interval between neighbour segments is obvious larger than the constant value, it may stand for several points are missed. The window area will rather show up as a data gap like this. And the occlusion relation is not existing in this situation anymore. Hence, this first constraint utilised for check whether there is occlusion relation between neighbour segments. We use threshold th_{angle} to define if these two segments have the gap in between. The second criterion is used to find if these segments are close to each other when considering segmentation process may generate over-segmentation result. However, the closet distance ($d_{closest}$) between neighbor points will be affected by the distance from centroid o to these points (l_i). And this is the reason that I use dynamic threshold $d_{dynamic}$ in this step. The distance between closest points p_j and p_j as shown in Figure 3-11. The normal vector of segment i and j is $\vec{V_i}$and $\vec{V_j}$ respectively. There are mainly four steps to achieve dynamic threshold $d_{dynamic}$.

1. Since the scanning angle around 0.625°. A triangle (as shown in Figure 15) can be built by expected scanning angle and line vector.
2. Calculate d1 and d2 based on the triangle.
3. Let two times of the mean values of d1 and d2 as the dynamic threshold $d_{dynamic}$. This is for move the effect of noise and system error.

When two segments satisfy these conditions, we extract the distance between centroid o and endpoint sj or si. Hence the segment with longer distance occluded by shorter one. The parameter used in this function shows in Table 6.

Table 6: Parameters used in the door detection function.

Parameter Name	Description
th_{angle}	The threshold value of scanning angle between neighbour points.

3.6.2.2. Parameter Analysis

The threshold value (th_{angle}) of scanning angle between neighbour points is also depend on the scanner. The angle between two points is 0.648° generally. In experiment, I use 3° as threshold. If we set a big value, it might see two separate segments as connected. In contrast, small threshold may detect too many scanning gaps in scanline.

3.6.3. Opening Detection

The opening is an essential component of indoor navigation and mapping. In this research, the aim is detecting opening based on scanlines analysis. However, we can use both specific criteria as well as candidate analysis to identify windows and doors in the future work. This approach is divided into two parts:

- Generation of opening candidates based on single scanline (scanner coordinate system).
- Determination of the optimal opening by including neighboring scanlines in the analysis (local coordinate system).

The openings, such as doors and windows, are usually seen as holes on a plane (Nikoohemat et al., 2017). Hence, the basic assumption is that opening area generally between two collinear segments in single scanline. The detection is started by finding a pair of collinear segments through a set of constraints and save the closest points in these extracted pair of segments as opening candidates. In order to analyse extracted opening candidates, we need project it to local coordinate system. The optimal openings are determined based on geometrical relations. This method could extract almost all windows, doors, either open or close, and work on real-time processing.

3.6.3.1. Opening Candidate Generation

At first, the parallel features (extracted from Section 3.5) of scanline were used to extract a pair of collinear segments. Figure 3-12 shows an example of door in horizontal scanline. As it can be observed that segment1 and 9 are collinear. If there are at least three segments are collinear, we only consider a pair of segment that do not have another collinear segment in between. Because it is not possible have other objects between the doorframe frame on the wall plane.

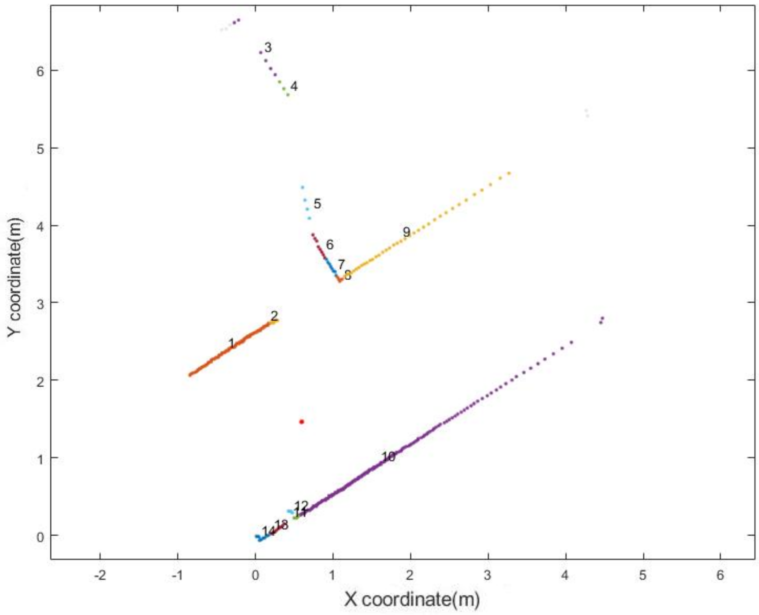


Figure 3-12: Detect door in the horizontal scanline (different colours label different segments, red point: scanner position)

The pattern of windows shows different characters with door. This is because the unfavourable material properties (Michailidis & Pajarola, 2016) of glass may miss the point cloud. However, in this dataset, when a laser beam penetrates window glasses (left window in Figure 3-13), some pluses will not be recorded by scanner when the distance beyond the measurement range, while some beams are reflected (the second window from right in Figure 3-13) when it bounces off the glass.

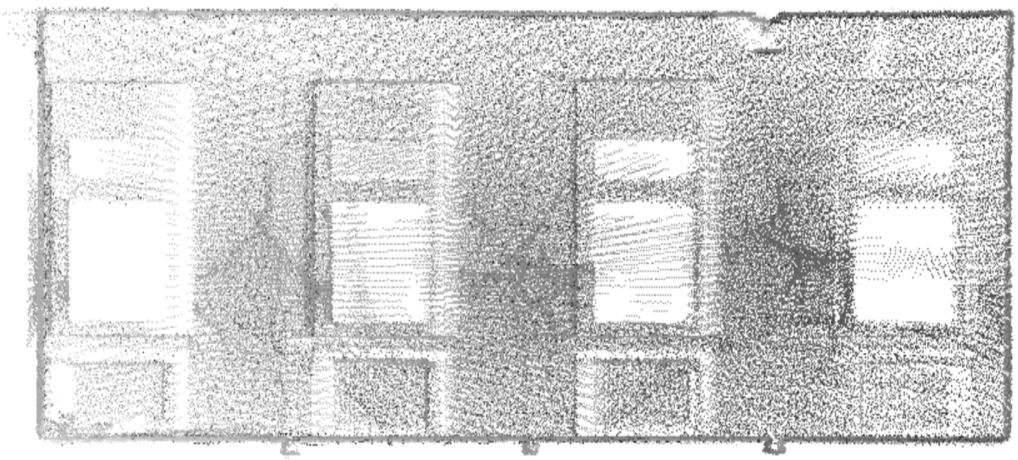


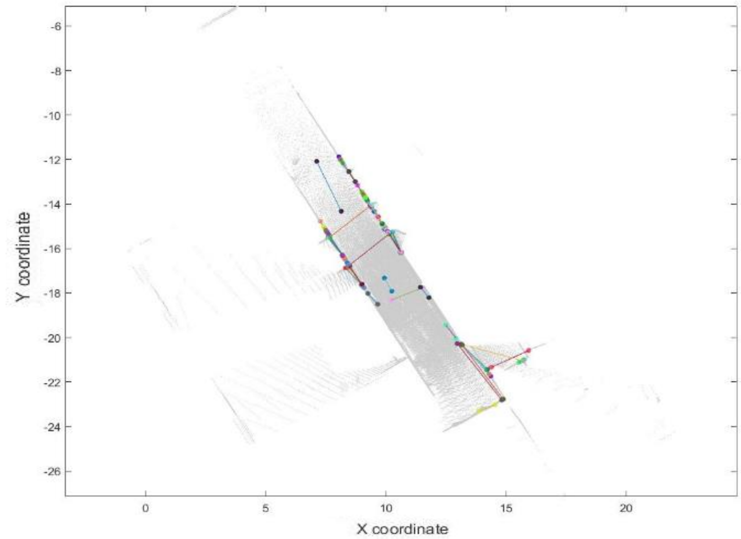
Figure 3-13: Example point cloud of windows

To extract opening candidates, these two situations should be considered. Once two collinear segments are extracted, a set of rules were used for find the opening segments. Let segment *i* and *k* on two sides of the opening area and line l_{ik} through these two segments.

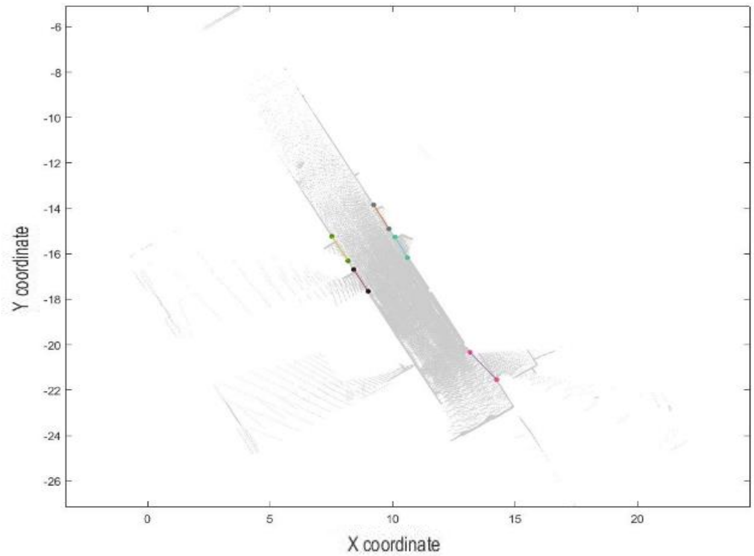
The rules of opening candidate extraction in single scanline are defined include the following:

- If *Si* and *Sk* are neighbour segment, probably these segments belongs to the window frame.
1. The scanning angle θ between these segments should lager than th_{angle}
2. The closest distance $d_{closest}$ between *Si* and *Sj* should larger than
- If there is at least one segment between segment *i* and segment *k,* probably these segments belongs to the door frame or there are points on the window
1. Find segment between segment *i* and *k*. Let d_{mean} be the mean Euclidean distance between the centroid of segment and the scanner position, d_{ik} be the mean distance between the closest endpoint of these two collinear segments and the scanner position. We found that the segment between the segment *i* and *k* should belong to another space. Hence, d_{mean} normally longer than d_{ik}.
2. The minimum distance of these segment between the two collinear segments should longer than d_{ik}.
3. The distance of the opening segments should longer than *dmin.*

If these conditions are fulfilled, we save closest endpoints $P_{door}1$ and $P_{door}2$ of this pair of segments as the opening candidates. Although opening can be extracted in single scanline, there is still some uncertainty. As it can be observed in Figure 3-14, it is obviously having some miss extracted points in the extracted points candidate, this is because certain scanlines may affected by the occlusion and clutter in indoor environment.



(a)



(b)

Figure 3-14: Top view of extracted opening candidates before (a) and after (b) improvement (different colour labelled different pair of points).

3.6.3.2. Optimal Opening Determination

In order to remove wrong points and save the segment in each opening area, we join the analysis of neighbour scanlines. We observed in Figure 3-14 (a) that the number of candidate point on doors or windows is obviously greater than the amount of miss extracted points. In xy-plane, the useful opening segments (defined by two endpoints) are generally close to each other. Consequently, we used following steps to find the opening segments.

As a rule of thumb, for the opening O: $\{O1, \ldots, Ok\}$ which has dozens of candidates, more than Nd of candidates within certain range is sufficient to define the location of doorframe.

1. Let point a and point b are the two endpoints of the opening segment. Since each side of the opening frame will be analysed in the following. It is important to find the point on the same side. I defined the point a be the point with smaller x coordinate, and point b is the bigger one. In order

to prevent the situation like two endpoints have same x coordinate. I set point a as the point have smaller y value, when the difference value of x coordinates smaller than the threshold. The extracted door frame points in the xy-plane of the local coordinate system as shown in Figure 3-15 .

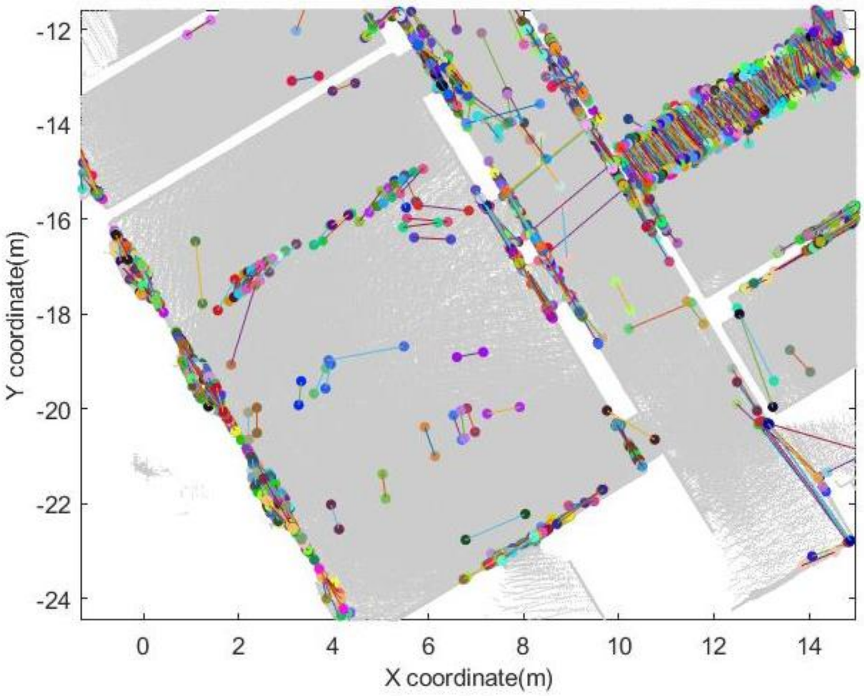


Figure 3-15. Example of detected opening candidates in XY-plane

2. Make a buffer with radius r_{circle_1} for all point a and point b. If more than *Nd* points are in same buffer and also for their corresponding point b, let the mean coordinates of point a and point b as the location of door frame. The doorframe segments will save by these two endpoints. This method will remove wrong point and remain the doorframe points, but the weakness is extract multiple doorframe points (like shown in Figure 3-16).

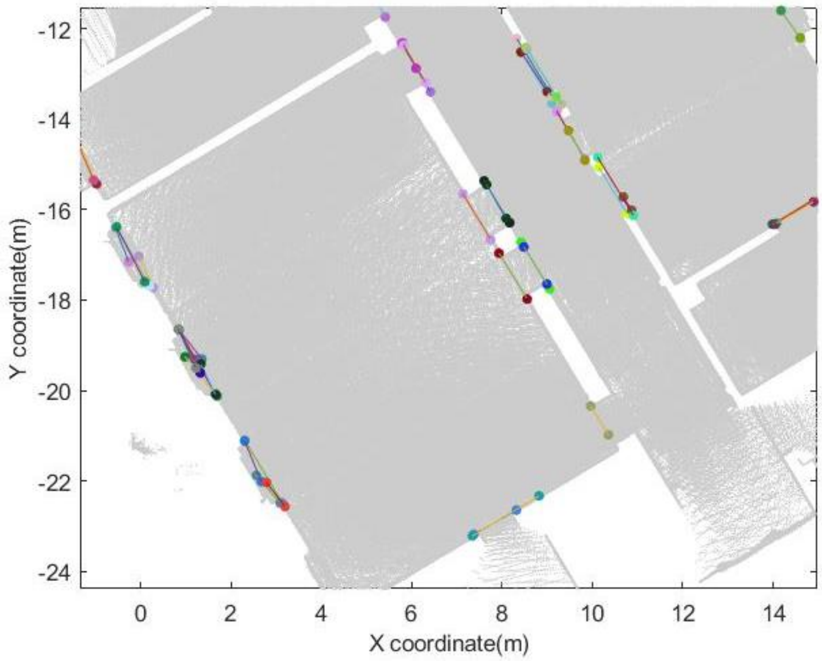


Figure 3-16. Example result of Step 2

3. To refine the result of Step 2, a new buffer with radius r_{circle_2} is used for merge the extracted doorframe points. An example as shown in Figure 3-17.

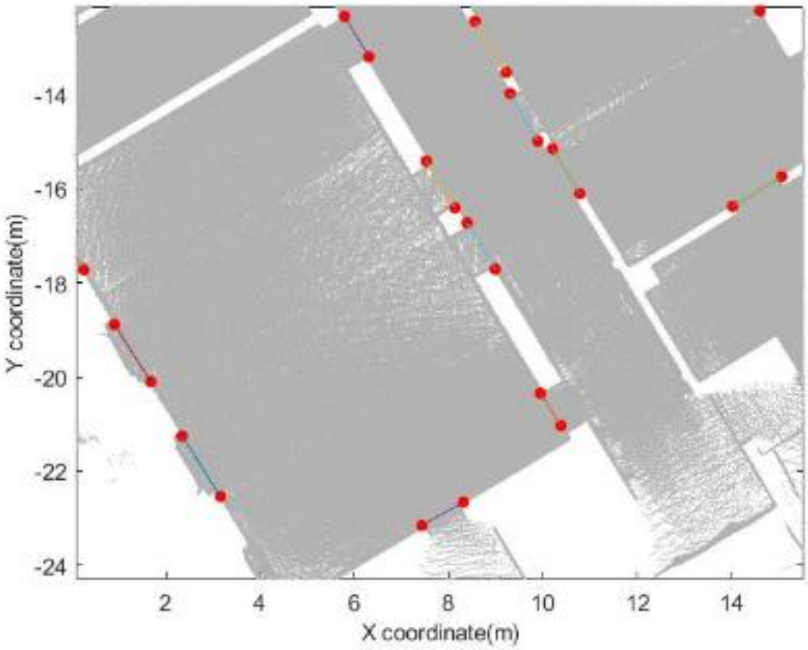


Figure 3-17. Example result of Step 3

4. Remove the detected opening that acquired by a number of point with low height variance. This is because the gap in the ceiling (The red circle in Figure 3-18) also meet the rules of opening candidate detection.

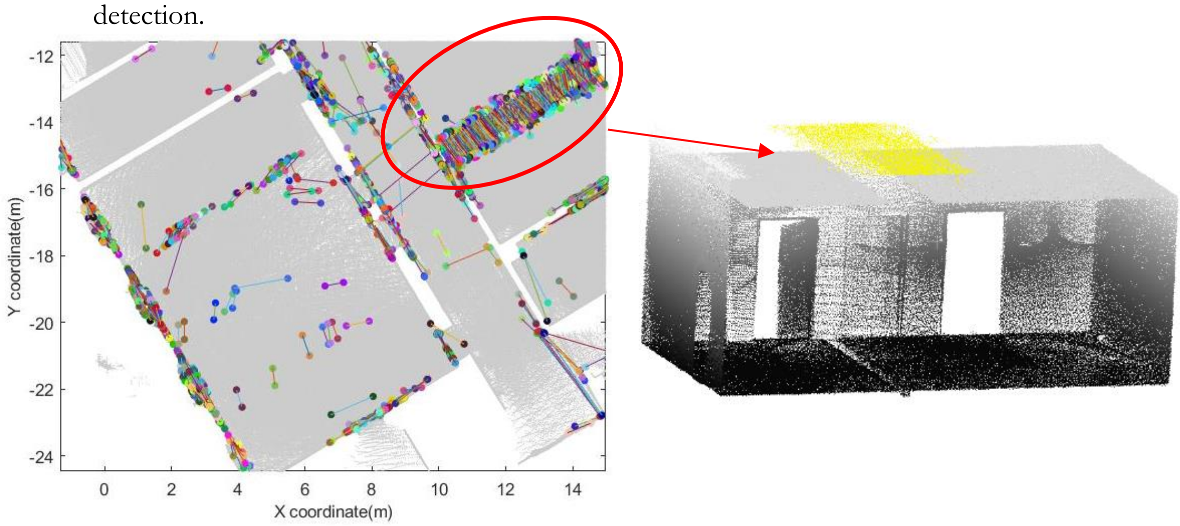


Figure 3-18: Effects of other special distributions.

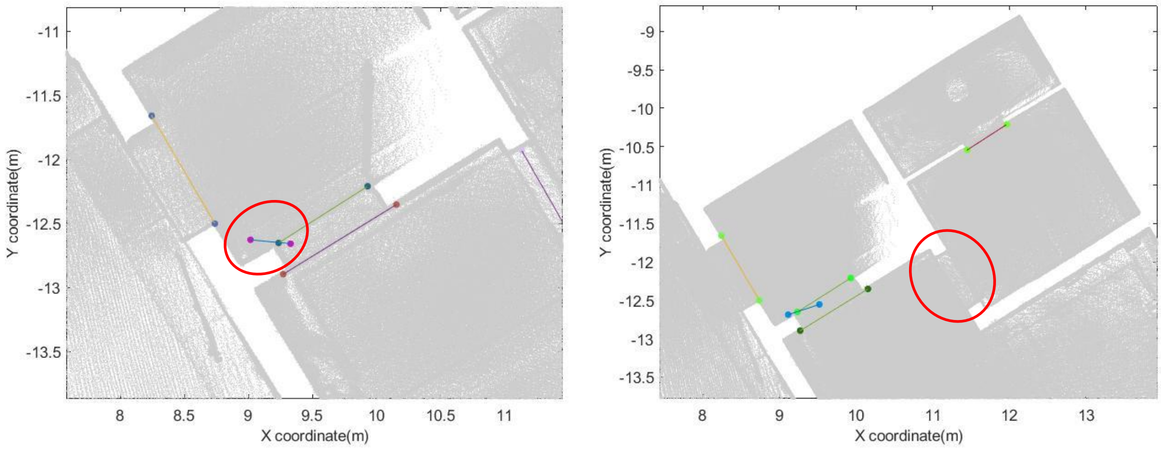
Table 7: Parameters used in the door detection function.

Parameter Name	Description
dmin	The minimum Euclidian distance between a pair of opening points.
r_{circle_1}, r_{circle_1},	The radius of finding circle.
Nd	The minimum number of segment to define the location of openings.

3.6.3.3. Parameter Analysis

The minimum Euclidian distance (*dmin*) between a pair of doorframe points will affect the detected doorframe points. We use *dmin*=0.7 m as the threshold based on experiment.

The radius (r_{circle_1}) of buffer and the minimum number (*Nd*) of the segment are important for door detection. If we set a big radius, it may merge these two doors together. The main object of step 2 is remove wrong door segment as well as merge the right door segments in opening detection. Meanwhile, big buffer size may lead more door segments in consideration, which means there are more possible effected by the wrong door candidates (an example as shown in the red circle in Figure 3-19 (a)). As for the minimum defined point number, small may remove opening area (an example as shown in the red circle in Figure 3-19 (b)). Hence, those two parameters were fixed to r_{circle_1}=0.3 and *Nd* =6 after the experiment. The parameter r_{circle_2} used to merge the segments that close to each other which set as 0.5(m) in this research.



(a) r_{circle}=0. 5 and *Ndoor*=6 (b) r_{circle}=0.3 and Ndoor=8

Figure 3-19: Influence of the parameter r_{circle_1} and *Nd*

3.6.4. Space Subdivision

A two-step subdivision process is defended in this Section. First, the point in the trajectory is subdivided into different space in combination with the detected opening segments. This process is working on xy-plane space, but it can be also generalized to 3D. Let k be the number of spaces labelling, x be the number of door frame segments. D_x denotes the list of distance between scanner's location and opening segments.

1. Input single scanline data and corresponding extracted features according in the order of time.
2. The step 1 is repeated for extract opening candidates each time.
3. Find the optimal openings based on extracted candidate.
4. The extracted openings are used to subdivide trajectory. As for trajectory point that has *th_space* away from the door's segment, we set this points as doorway.
5. When trajectory passed through a new opening segment, we increment the value of space, and label the new space as the incremented space value. This opening segment is then defined as doorframe segment.

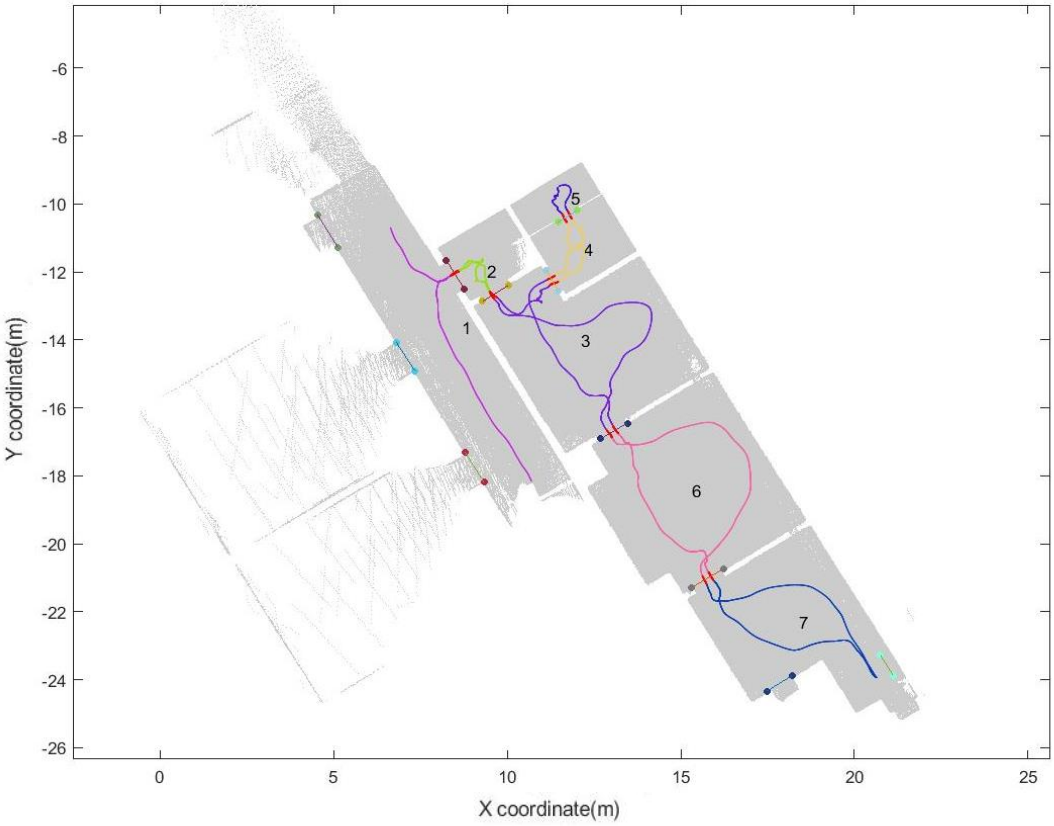
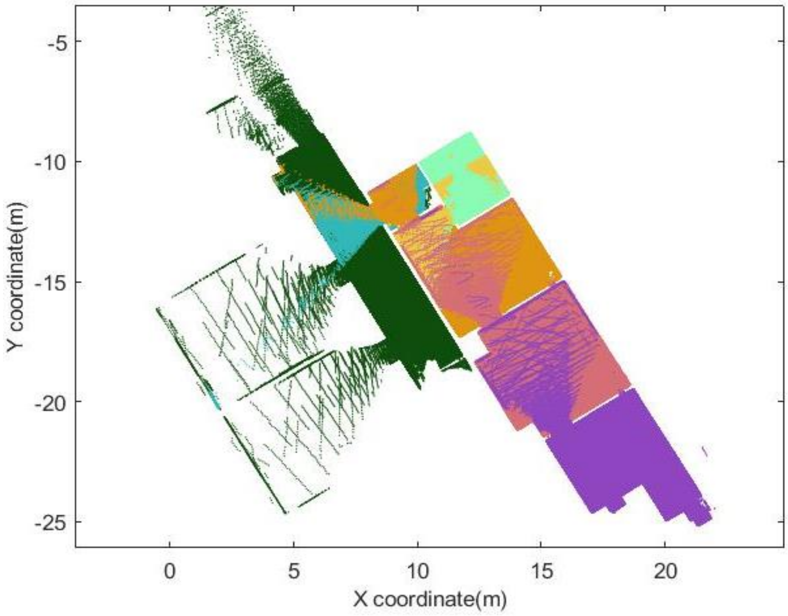


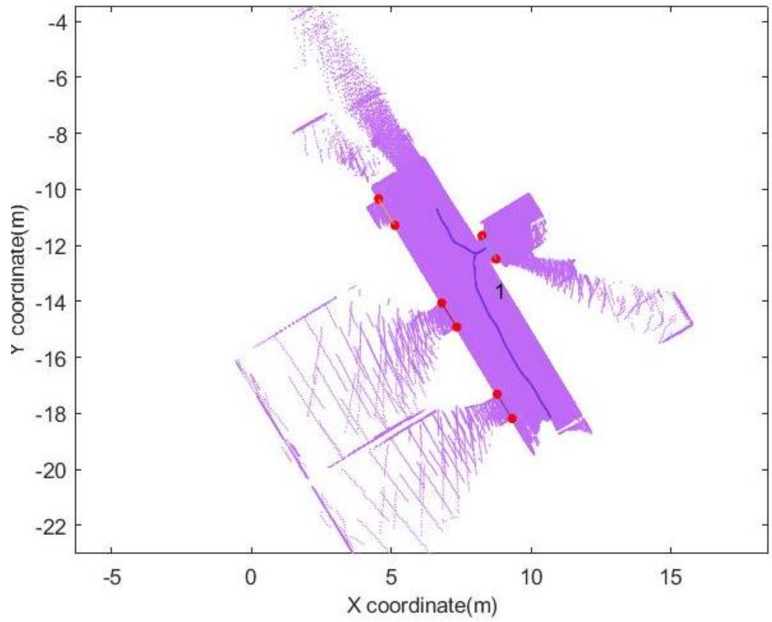
Figure 3-20: Example of trajectory subdivision result (different colours stand for different spaces).

After trajectory subdivision process, the point cloud can be subdivided based on the defined doors.

1. The point clouds subset is extract points based on the trajectory within same space, like in Figure 3-21 (a). This process is used to speed up the subsequent process.
2. As for each point cloud subset, we use the defined door to subdivide point cloud. This subdivide method based on the scanlines. The basic assumption is that if the point belongs to the space, the line links the scanner position and the point will not intersect with door segment in xy-plane. Therefore, we link the scanner location with each point of the scanline in xy-plane. If the line intersects with defined door segments, then the point label as the label of another space.



(a)



(b)

Figure 3-21: Example of the subset of point cloud. (a) all point cloud subset, (b) one of the point cloud subset.

4. RESULTS AND DISCUSSION

4.1. Dataset Description

The methodology of this research is tested in a real dataset, which includes both the 3D point clouds and the trajectory that collected by ZEB-REVO from GeoSLAM Company during the acquisition. The 3D point clouds were processed from raw laser range measurement and inertial data by using GeoSLAM's SLAM algorithm. Point clouds file and trajectory file are normally related by timestamp, which meaning that from each position in the trajectory we not only know the pose of scanner but which single scanline in the dataset were acquired (Díaz-Vilariño et al., 2017).

The two datasets that are tasted in this research was captured in one of the buildings of the Technical University of Braunschweig, Germany. A visualization of the two datasets can be seen in Figure 4-1. Dataset1 contains 7 rooms, 11 doors, both open and closed. Dataset 2 shows a long corridor and some rooms with windows. It is notice that operator did not go into some rooms in both of these datasets.

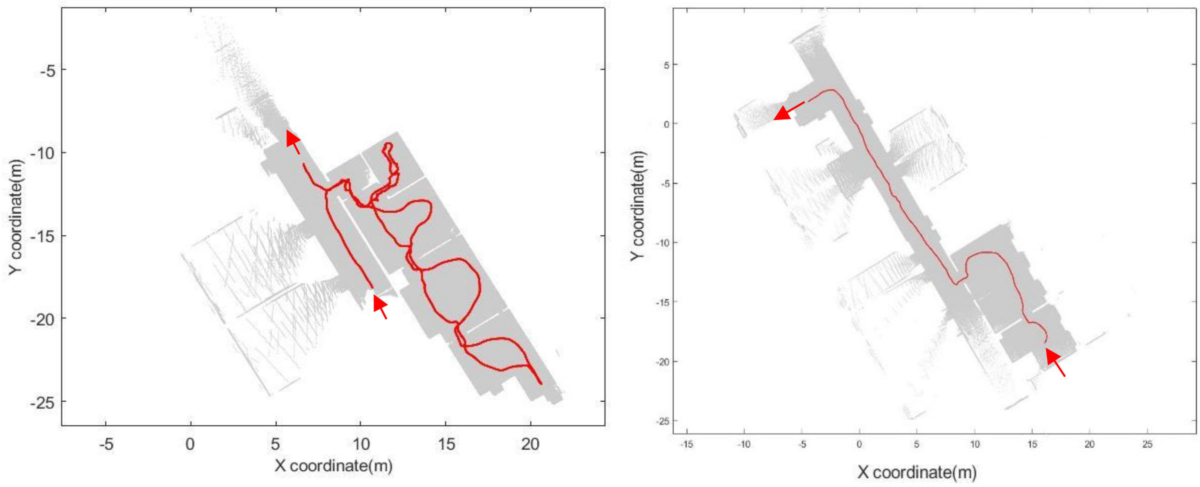


Figure 4-1: Top-down view of tasted dataset: dataset 1(left) and dataset 2 (right) (grey: point cloud; red: trajectory).

4.2. Preprocessing Result and Discussion

4.2.1. Scanlines Generation Results

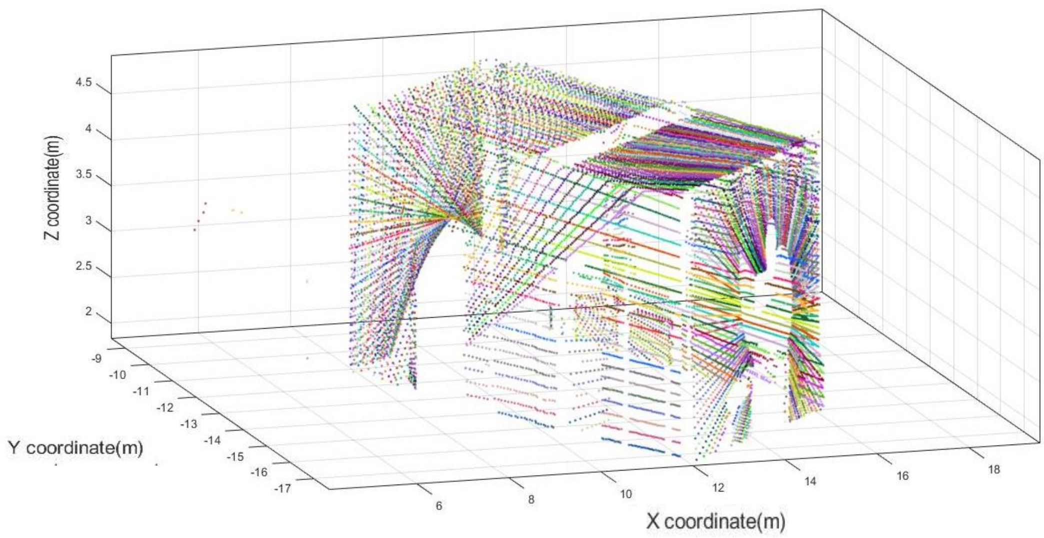


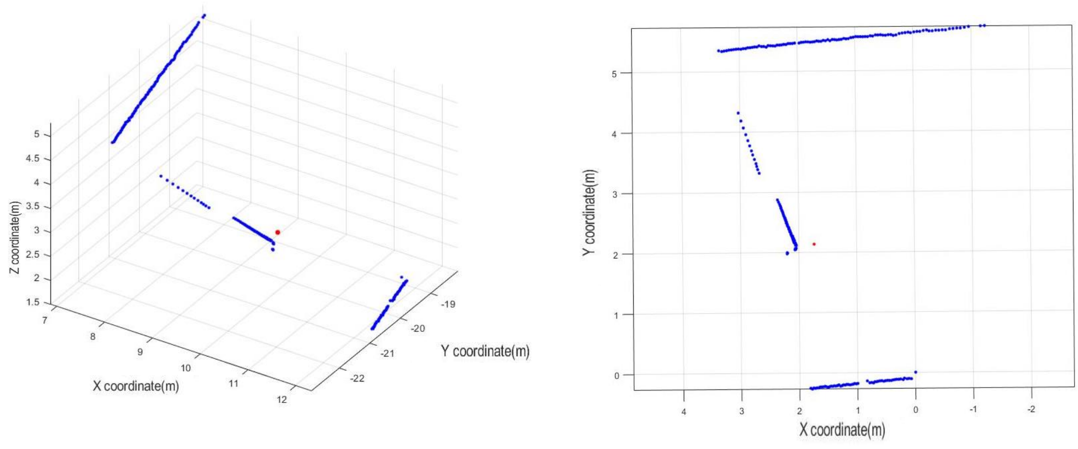
Figure 4-2: Example of extracted scanlines (difference scanline labelled different colour).

Table 8: The mean and variance of the Eigen value of the scanlines

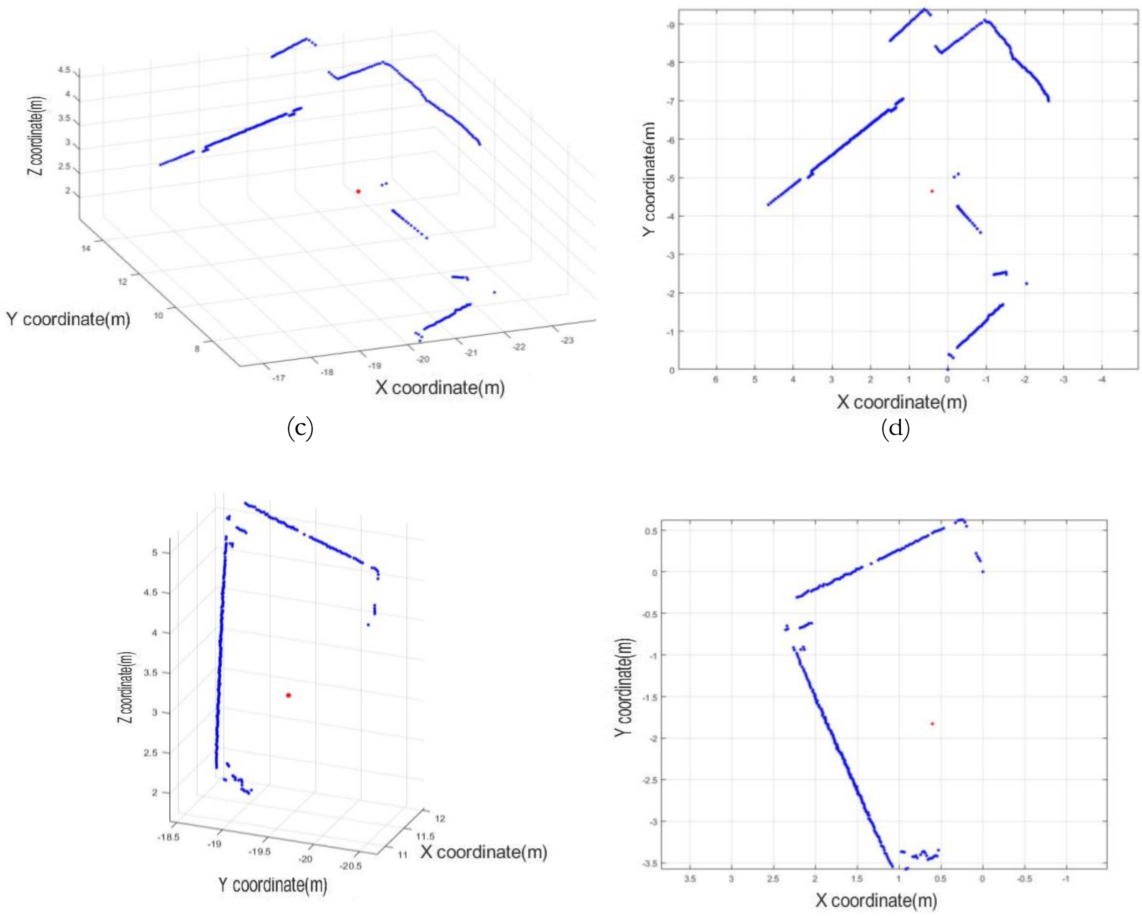
	Dataset 1			Dataset 2		
Eigen value	*V1*	*V2*	*V3*	*V1*	*V2*	*V3*
Mean	1.31e-05	0.78	2.02	3.05e-05	2.17	4.92
Variance	9.73e-10	3.29	3.29	2.56e-09	1.75	10.32

Figure 4-2 shows a part of scanline generation result in dataset 2. It ensures that the scanlines are distinctly separated even in planes of overlap. Table 8 shows the mean and variance of the Eigen value. As in illustrated in Section 3.3.1, the variance along the plane normal is estimated by smallest eigenvalue. Hence $\lambda1$ is used in this part to check the quality of the scanline. The result shows that almost all the extracted scanlines lie on the same plane, it almost does not have under-extracted scanlines.

4.2.2. Projection Results



(a) (b)



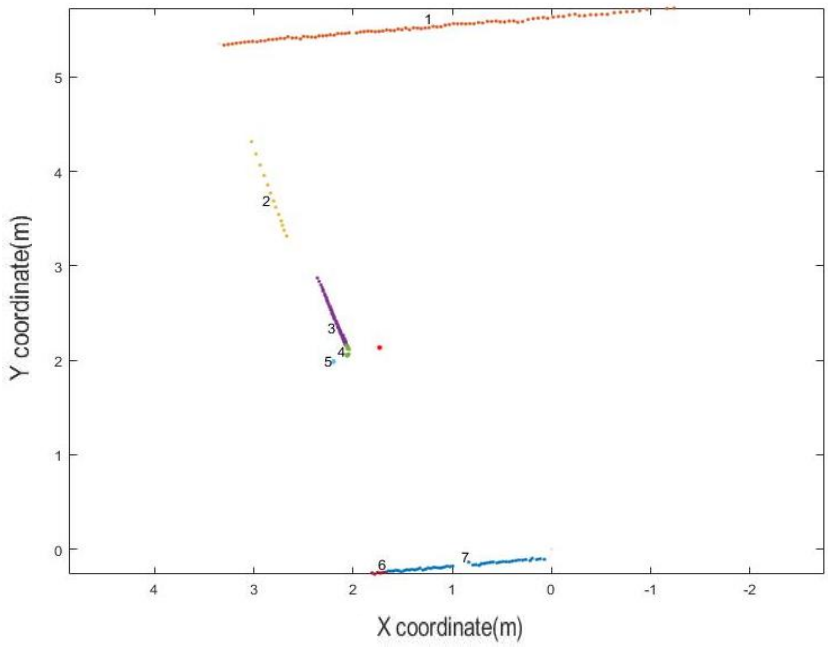
(c) (d)

(e) (f)

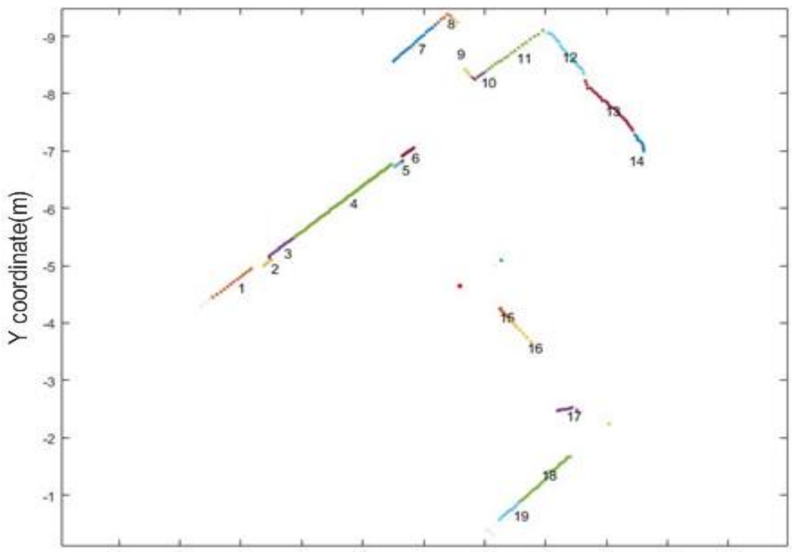
Figure 4-3: The resulting 2D scanlines where comparing with their corresponding 3D scanlines in terms of different roll angels ((a)(b): Roll = -37.7209, (c)(d): Roll = -154.2332, (e)(f): Roll = 91.9130).

4.3. Segmentation Results

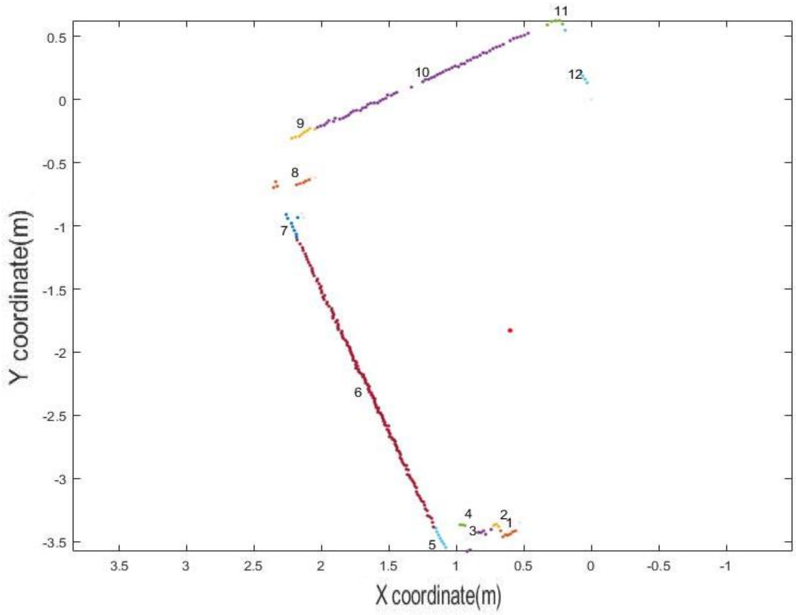
In order to evaluate the performance of segmentation algorithm, I have tasted 50 scanlines acquired from different location and roll angel on both datasets. Figure 4-4 (a) to (c) show the processing results of segmentation in a variety of roll angles.



(a) Roll = -37.72°



(b) Roll = -154.23°

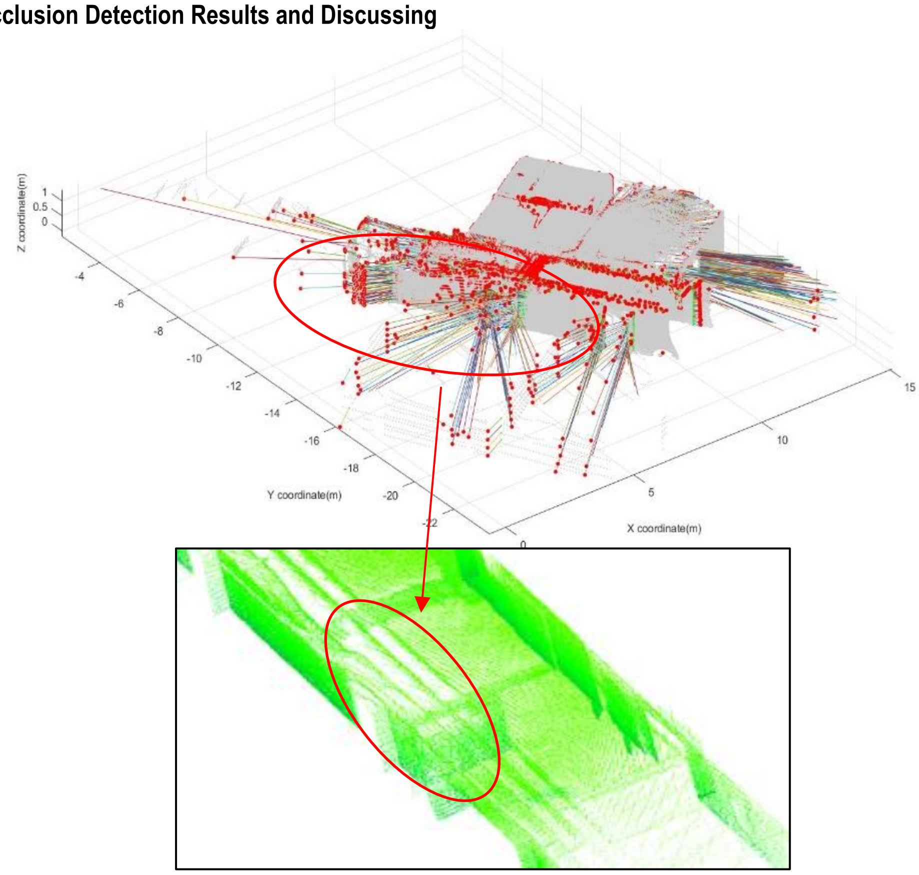


(c) Roll = 91.9130°

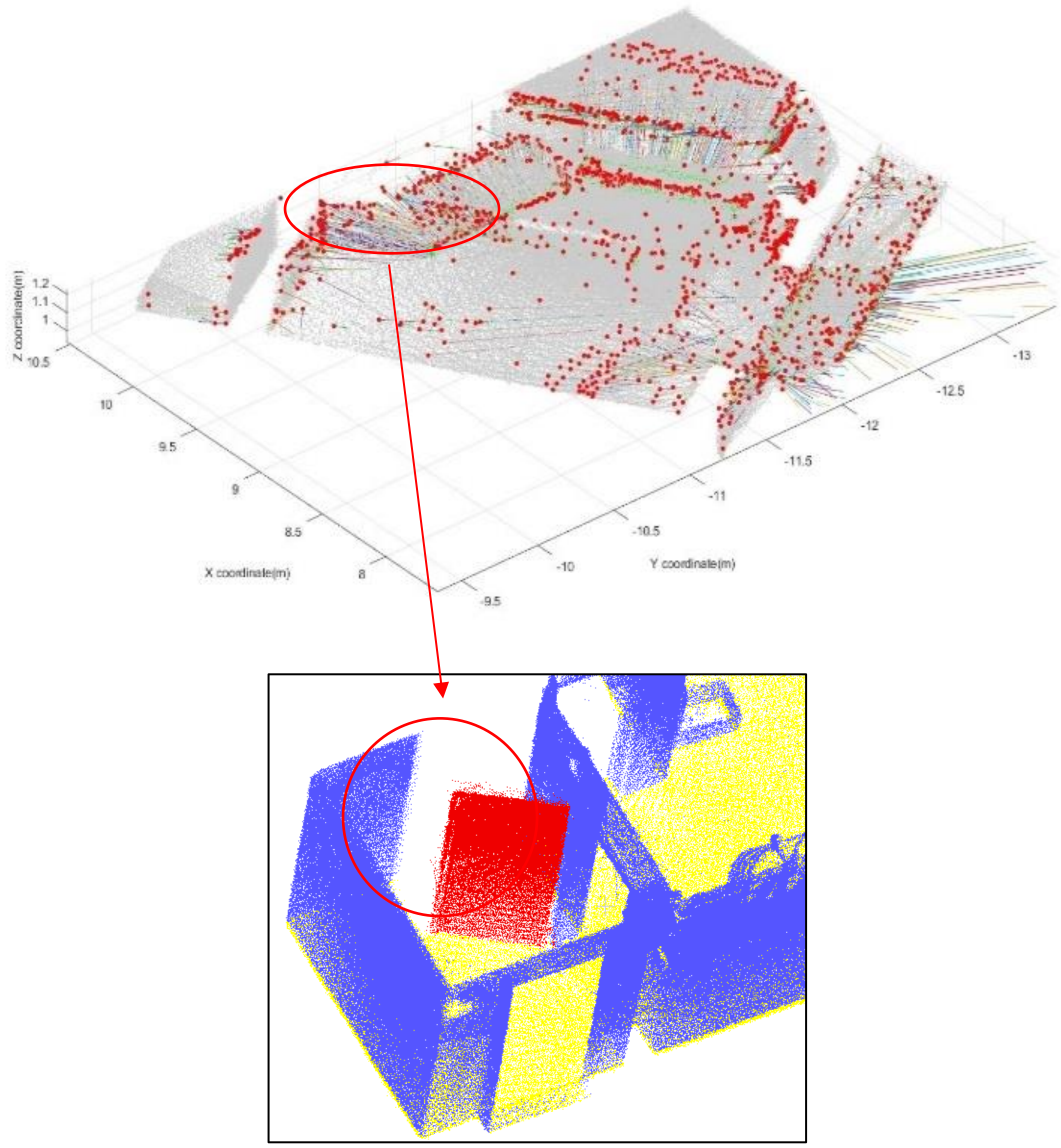
Figure 4-4: Results produced by segmentation based on the scanlines with different roll angle. It should be notice that these results are produced based on the best parameter combination (as shown in Table 4).

4.4. Classification Result and Discussion

4.4.1. Occlusion Detection Results and Discussing



(a)

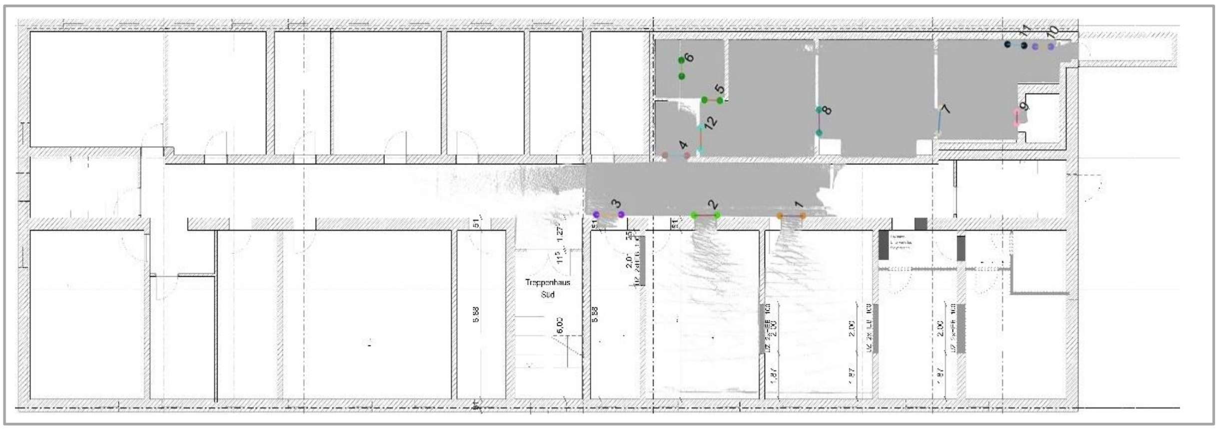


(b)

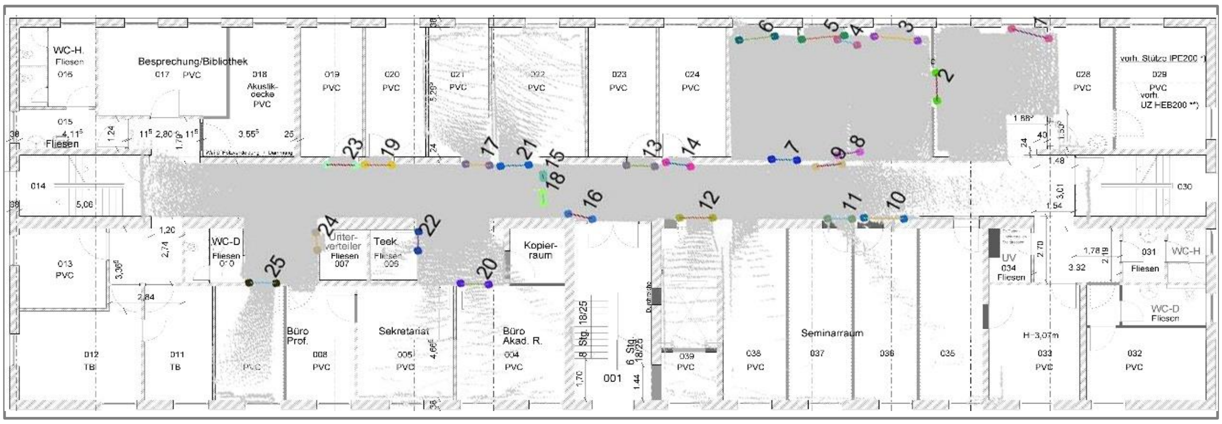
Figure 4-5: The occlusion results in a part of the dataset.

The segment between the endpoints of the segments that have occlusion relation were used for visualise occlusion relations. Figure 4-5 (a) presents numerous occlusion relations on the ceiling. And the corresponding point clouds shows there are some pipe lines close to the ceiling. Figure 4-5 (b) showcase an occlusion example between door (red plane) and wall (blue plane).

4.4.2. Door detection results and discussion



(a)



(b)

Figure 4-6: Visual comparison of door extraction results and ground plan in dataset 1 and dataset 2. (a) extracted doors in dataset 1 (b) extracted doors in dataset 2 (doorframe segments are shown in different colours).

Using the manually registered ground plan as reference, a visual analysis is carried out to check if the extracted doors are extracted correctly and completely:

- In the Figure 4-6 (a), most of the extracted door segments match the corresponding doors in ground plan. However, door 1 and 2 do not match the door location so well (as shown in Figure 4-7). This can be explained by two reasons: the error in SLAM results or the wrong ground plan.

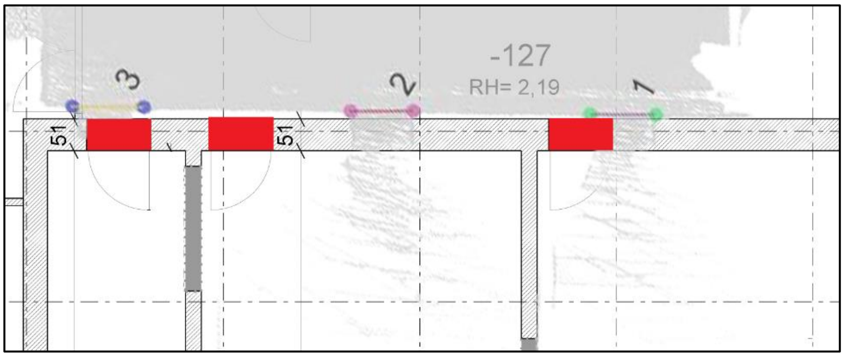
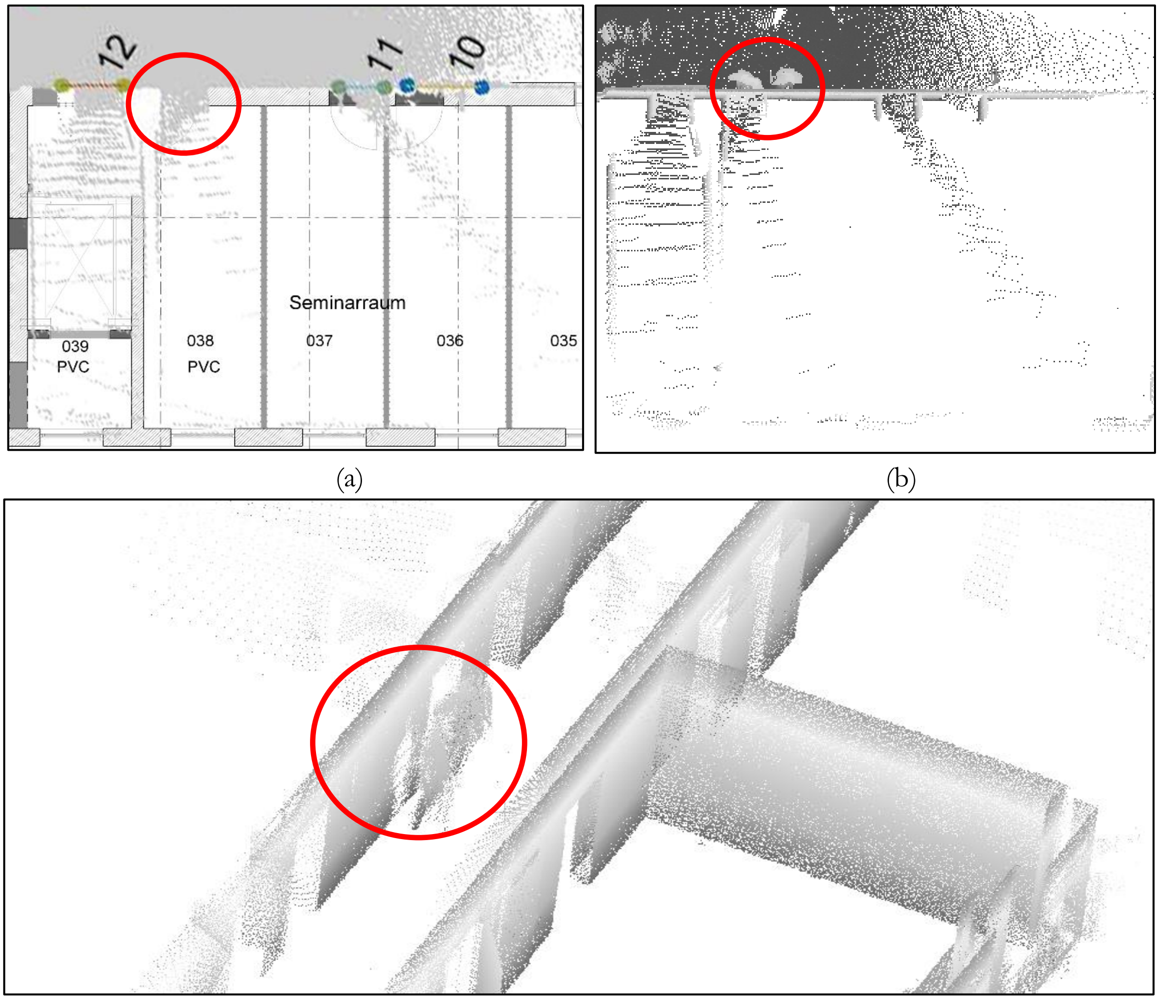


Figure 4-7. Comparison of results obtained opening detection and ground plan (detail in Figure 4-6 (a), red: door location in ground plan).

- It can be observed that the doors in both of the datasets have high detection accuracy, even for some doors (such as door 19 and 20 in Figure 4-6 (b)) closed to each other.

- This method can also detect closed door (like the door 3,9 in Figure 4-6 (a) and the door 13,14,19 and 23 in Figure 4-6(b)). But it is noticed the closed-door detection based on the door's geometry so it cannot detect some closed-doors that are co-planar to the wall.
- The door 9 in Figure 4-6 (b) did not fit well with the wall. Multiple reasons may let this result. First is the poor segmentation result. For example, if the points in corner are discarded in segmentation process, the accurate acquired point will not be used in opening detection. Second is sparse point in scanline.
- The door between door 11 and 12 did not be detected. This is because the door is occluded by another object. Hence, the door does not show as the expected pattern in single scanline.



(a) (b)

(c)

Figure 4-8. The dis-detected door in ground plane(a), xy-plane(b) and 3D point clouds (detail in Figure 4-6 (b)).

- Interestingly, the double door in the corridor area is extracted as door 15 and 18 in Figure 4-6 (b) after compared with ground plan. The reason why the double door is detected as two doors is the double door shows as one open door and one closed door (as shown in Figure 4-9) during data acquisition.

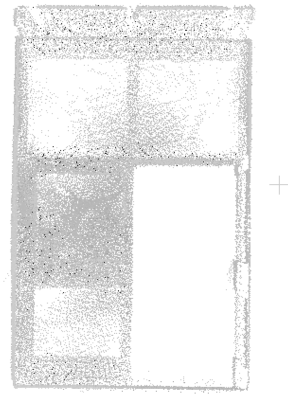
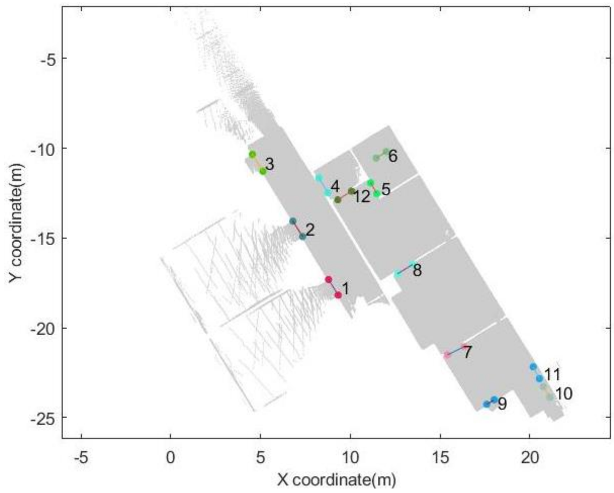


Figure 4-9: Point cloud of the double-door in corridor area.



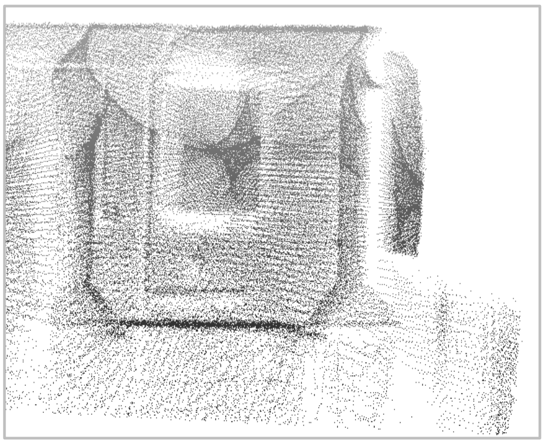
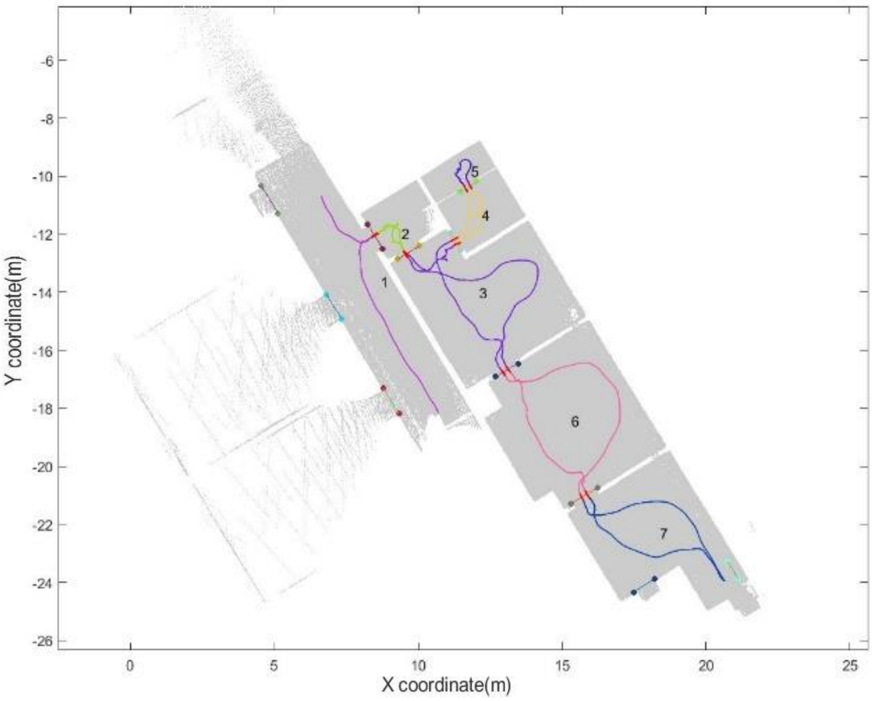


Figure 4-10: Compare detection result (left) with point clouds (right).

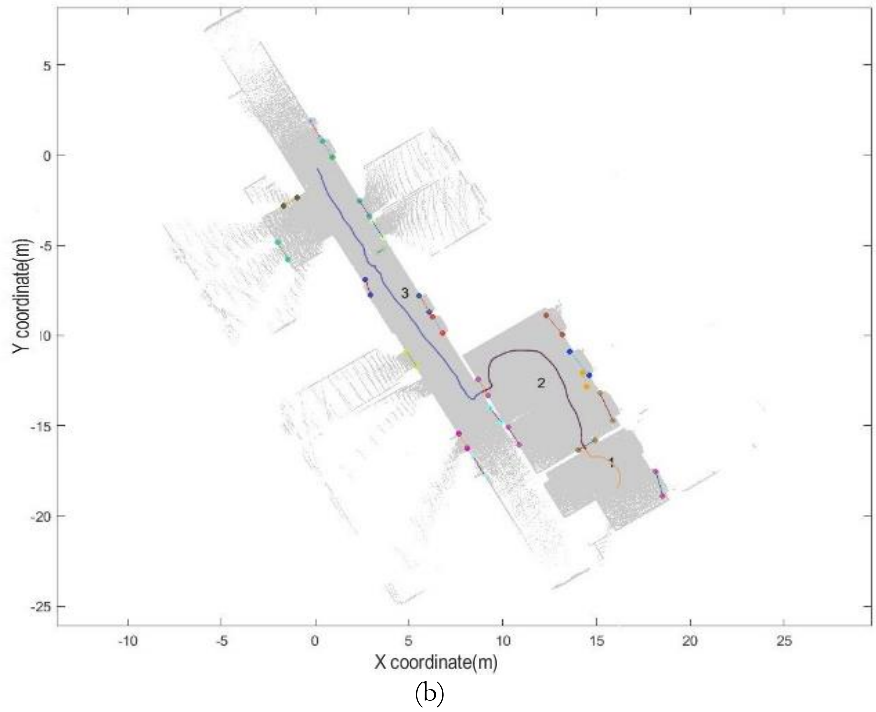
The dataset 1 is acquired in basement, hence it not possible detect openings seems like window in point cloud. The corresponding point clouds (as shown in Figure 4-10 (right)) shows some part of this wall is concaved. This error demonstrates the door detection method may affected by some indoor objects that have the geometric structure similar with openings.

4.4.3. The subdivided trajectory results

4.4.3.1. Trajectory subdivision



(a)

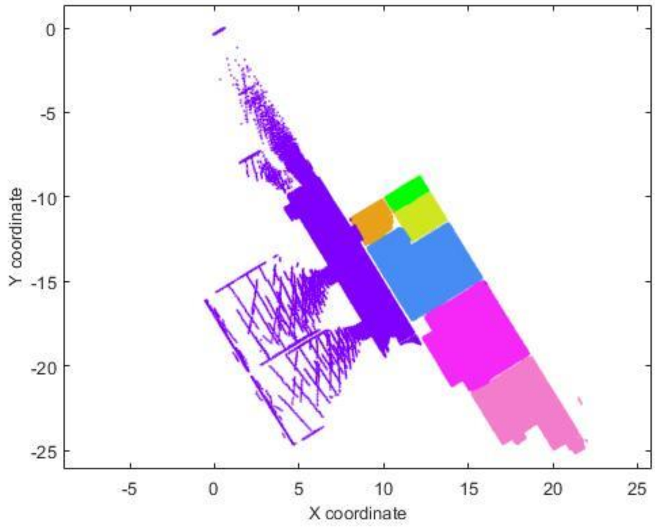


(b)

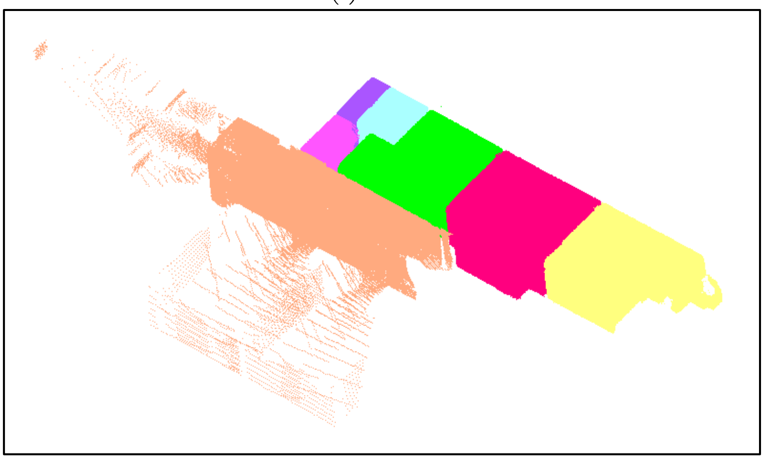
Figure 4-11: The visualization of subdivided trajectory results in dataset 1(a) and dataset 2 (b).

Figure 4-11 demonstrate the good trajectory subdivision results in both datasets. However, this method will produce wrong result when operator go into a space through one door and left this space from another room. This is because the method saves the label of space based on the door's location.

4.4.3.2. Space Subdivision Results



(a)



(b)

Figure 4-12: Space subdivision result in 2D (a) and 3D (b) view.

5. CONCLUSION AND RECOMMENDATIONS

The increasing quality of IMLS data has triggered this research on the suitability of object detection based on scanline analysis. This thesis investigated the possibility to detect objects based on using scanlines in IMLS data. The general conclusion is given in Section 5.1. After that, all the research questions are revisited in Section 5.2. Finally, based on Section 5.3 outlines the recommendation based on the conclusion and the research questions.

5.1. Conclusions

In this study, the novel method is designed to detect openings as well as subdivide space based on scanline analysis. In order to work on the real-time processing, the detection method first generate the candidate based on the single scanline. As the scanline number increase during data acquisition, the number of the candidates is growing accordingly. After that, a determination method is used for extracting the optimal results. After visual comparison of the openings results and the ground plane, the detection method has demonstrated the validity of the algorithm for detecting both open and closed doors. Another research object is space subdivision. The space subdivision method was split into two steps. Firstly, the trajectory points are subdivided into different spaces. Secondly, the corresponding point clouds that related to the trajectory points within same space are subdivided based on the detected doors. In the end, openings information and space subdivision results are saved as point cloud label, which will be used for further investigations.

5.2. Answer to Research Questions

1. Which objects can be extracted and detected in single scanline?
   The classification method is based on analysis geometric structure of single scanline. In this research, opening shows diacritical structure in scanline steady. Hence, it can be detected based on single scanline. But as for some indoor object, like curtain, may shows different pattern or multiple short segments in scanline, is hard extracted in single scanline.
2. Which features can be utilized in detection processing?
   There are nine features of segments have been proposed based on analysis geometric and topological properties.
3. What is the distribution of the noise and how can it be removed?
   The noise normally random distributed in the point clouds. However, it shows as the point or short segment in the scanline. These noises were removed by defined the minimum number of point to form the reliable segments in the segmentation process.
4. Will the adjacent scanline beneficial to recognize indoor objects? If it is, how many?
   The adjacent scanline in this research is defined based on the time attribute. In the opening detection method, the adjacent scanlines were used for acquire opening candidates. However, the number of adjacent scanline cannot be determined. This is because the behavior of the operator during data acquisition will affect the number of the neighbor scanlines. Hence, it has different number of neighbor scanlines when detect different openings.

5. How to use adjacent scanlines to improve detection results?
   The adjacent scanlines provide more opening candidates in opening detection, which is helpful to remove some wrong extracted candidates and determine the optimal openings. In this research, I used an optimization method to analysis the candidates that extracted from adjacent scanlines for improve the detection results.
6. Are the trajectory and extracted objects useful to space subdivision?
   The space subdivision method is based on the extracted doors. However, if we only use the extracted doors to subdivide space, it may lead high computational load and low processing speed because they need analysis numerous points. Trajectory is used to solve this problem. First of all, trajectory is easy to be subdivide by the detected doors based on distance from door segment and scanner position. Meanwhile, trajectory and point cloud are related by timestamp. Hence, the point clouds can be divided into several subsets based on corresponding subdivided trajectory. This step could reduce the processing time as well as computational load.
7. How to asset the experiments result?
   The ground plan has been used to evaluate the detection results.

5.3. Recommendation

The recommendations are listed as follows:

1. Occlusion detection
   The occlusion detection proposed a possibility that generates the occlusion relations based on scanline, which could be used in indoor reconstruction in the future. In this research, the occlusion relations are only detected based on the single scanline, which is uncertainty. The future work will focus on combine all occlusion candidate in consideration.
2. Openings detection
   Although the experiment shows good result in opening detection, it cannot separate doors and windows so far. The future work will focus on analysis the distribution of the opening candidates to detected doors and windows. In order to achieve this goal, the maximum, minimum and variance of z-coordinate value of the candidates will be analysis. Unlike doors, windows generally have a distance to the floor. Meanwhile, windows have lower height variance than doors. Hence, doors and window can be detected based on these assumptions. However, this assumption not work for windows to the floor.
3. Space subdivision
   Even opening detection process can extract almost all of the windows and doors in the space, the door that not intersect with trajectory cannot be extracted as ‘door’ from the openings. If the doors and windows can be identified directly in the future work, it is possible to improve the space subdivision results.

LIST OF REFERENCES

Agapito, L., Bronstein, M. M., & Rother, C. (2015). Computer Vision - ECCV 2014 Workshops: Zurich, Switzerland, September 6-7 and 12, 2014 Proceedings, Part IV. In L. Agapito, M. M. Bronstein, & C. Rother (Eds.), *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (Vol. 8928). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-319-16220-1

Aijazi, A. K., Serna, A., Marcotegui, B., Checchin, P., & Trassoudaine, L. (2016). Segmentation and classification of 3D urban point clouds: Comparison and combination of two approaches. In *Springer Tracts in Advanced Robotics* (Vol. 113, pp. 201–216). https://doi.org/10.1007/978-3-319-27702-8_14

Buck U, Naether S, Räss B, Jackowski C, T. M. (2013). Accident or homicide – Virtual crime scene reconstruction using 3D methods, *225*, 75–84. https://doi.org/10.1016/j.forsciint.2012.05.015

Díaz-Vilariño, L., Verbree, E., Zlatanova, S., & Diakité, A. (2017). Indoor modelling from SLAM-based laser scanner: Door detection to envelope reconstruction. In *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives* (Vol. 42, pp. 345–352). https://doi.org/10.5194/isprs-archives-XLII-2-W7-345-2017

Dos Santos, D. R., Basso, M. A., Khoshelham, K., De Oliveira, E., Pavan, N. L., & Vosselman, G. (2016). Mapping Indoor Spaces by Adaptive Coarse-to-Fine Registration of RGB-D Data. *IEEE Geoscience and Remote Sensing Letters*, *13*(2), 262–266. https://doi.org/10.1109/LGRS.2015.2508880

Grilli, E., Menna, F., & Remondino, F. (2017). A REVIEW OF POINT CLOUDS SEGMENTATION AND CLASSIFICATION ALGORITHMS. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, *XLII-2/W3*, 339–344. https://doi.org/10.5194/isprs-archives-XLII-2-W3-339-2017

Hebel, M., & Stilla, U. (2008). Pre-classification of points and segmentation of urban objects by scan line analysis of airborne LiDAR data. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, *37*, 105–110. Retrieved from http://www.isprs.org/proceedings/XXXVII/congress/3_pdf/16.pdf

Henry, P., Krainin, M., Herbst, E., Ren, X., & Fox, D. (2010). RGB-D Mapping : Using Depth Cameras for Dense 3D Modeling of Indoor Environments. *RGBD Advanced Reasoning with Depth Cameras Workshop in Conjunction with RSS*, *1*(c), 9–10. https://doi.org/10.1177/0278364911434148

Holopainen, M., Kankare, V., Vastaranta, M., Liang, X., Lin, Y., Vaaja, M., … Alho, P. (2013). Tree mapping using airborne, terrestrial and mobile laser scanning – A case study in a heterogeneous urban forest. *Urban Forestry & Urban Greening*, *12*, 546–553. https://doi.org/10.1016/j.ufug.2013.06.002

Hu, X., & Ye, L. (2013). a Fast and Simple Method of Building Detection From Lidar Data Based on Scan Line Analysis, *II*(May), 7–13. Retrieved from https://www.isprs-ann-photogramm-remote-sens-spatial-inf-sci.net/II-3-W1/7/2013/isprsannals-II-3-W1-7-2013.pdf

Ishikawa, R., Roxas, M., Sato, Y., Oishi, T., Masuda, T., & Ikeuchi, K. (2016). A 3D reconstruction with high density and accuracy using laser profiler and camera fusion system on a rover. In *Proceedings - 2016 4th International Conference on 3D Vision, 3DV 2016* (pp. 620–628). https://doi.org/10.1109/3DV.2016.70

Kerl, C., Souiai, M., Sturm, J., & Cremers, D. (2015). Towards illumination-invariant 3D reconstruction using ToF RGB-D cameras. *Proceedings - 2014 International Conference on 3D Vision, 3DV 2014*, 39–46. https://doi.org/10.1109/3DV.2014.62

Khoshelham, K., & Elberink, S. O. (2012). Accuracy and resolution of kinect depth data for indoor mapping applications. *Sensors*, *12*(2), 1437–1454. https://doi.org/10.3390/s120201437

Klepeis, N. E., Nelson, W. C., Ott, W. R., Robinson, J. P., Tsang, A. M., Switzer, P., … Engelmann, W. H. (2001). The National Human Activity Pattern Survey (NHAPS): a resource for assessing exposure to environmental pollutants. *Journal of Exposure Analysis and Environmental Epidemiology*, *11*(3), 231–252. https://doi.org/10.1038/sj.jea.7500165

Koppula, H. S., Anand, A., Joachims, T., & Saxena, A. (2011). Semantic Labeling of 3D Point Clouds for Indoor Scenes. *Neural Information Processing Systems*, 1–9.

https://doi.org/10.1109/CVPR.2012.6247992
Kukko, A., Kaartinen, H., Hyyppä, J., & Chen, Y. (2012). Multiplatform mobile laser scanning: Usability and performance. *Sensors (Switzerland)*. https://doi.org/10.3390/s120911712
Lin, C. H., Chen, J. Y., Su, P. L., & Chen, C. H. (2014). Eigen-feature analysis of weighted covariance matrices for LiDAR point cloud classification. *ISPRS Journal of Photogrammetry and Remote Sensing*, *94*, 70–79. https://doi.org/10.1016/j.isprsjprs.2014.04.016
Luo, C., & Sohn, G. (2013). Line-based Classification of Terrestrial Laser Scanning Data using Conditional Random Field. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, *XL-7/W2*(November), 155–160. https://doi.org/10.5194/isprsarchives-XL-7-W2-155-2013
Mattausch, O., Panozzo, D., Mura, C., Sorkine-Hornung, O., & Pajarola, R. (2014). Object detection and classification from large-scale cluttered indoor scans. *Computer Graphics Forum*, *33*(2), 11–21. https://doi.org/10.1111/cgf.12286
Michailidis, G.-T., & Pajarola, R. (2016). Bayesian graph-cut optimization for wall surfaces reconstruction in indoor environments. *The Visual Computer*, 1–9. https://doi.org/10.1007/s00371-016-1230-3
Mura, C., Mattausch, O., Jaspe Villanueva, A., Gobbetti, E., & Pajarola, R. (2014). Automatic room detection and reconstruction in cluttered indoor environments with complex room layouts. *Computers and Graphics (Pergamon)*, *44*(1), 20–32. https://doi.org/10.1016/j.cag.2014.07.005
Nan, L., Xie, K., & Sharf, A. (2012). A Search-Classify Approach for Cluttered Indoor Scene Understanding. *ACM Trans. Graph. Article*, *31*(10). https://doi.org/10.1145/2366145.2366156
Nguyen, V., Martinelli, A., Tomatis, N., & Siegwart, R. (2005). A comparison of line extraction algorithms using 2D laser rangefinder for indoor mobile robotics. In *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems* (pp. 1929–1934). IEEE. https://doi.org/10.1109/IROS.2005.1545234
Nikoohemat, S., Peter, M., Oude Elberink, S., & Vosselman, G. (2017). EXPLOITING INDOOR MOBILE LASER SCANNER TRAJECTORIES FOR SEMANTIC INTERPRETATION OF POINT CLOUDS. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, *IV-2/W4*, 355–362. https://doi.org/10.5194/isprs-annals-IV-2-W4-355-2017
Ochmann, S., Vock, R., Wessel, R., Tamke, M., & Klein, R. (2014). Automatic Generation of Structural Building Descriptions from 3D Point Cloud Scans. *Proceedings of GRAPP 2014 - International Conference on Computer Graphics Theory and Applications*, *January*. Retrieved from http://cg.cs.uni-bonn.de/aigaion2root/attachments/GRAPP_2014_54_CR.pdf
Peter, M., Fritsch, D., Schafer, B., Kleusberg, A., Link, J. A. B., & Wehrle, K. (2012). Versatile geo-referenced maps for indoor navigation of pedestrians. *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, (November), 1–4. Retrieved from http://www.comsys.rwth-aachen.de/fileadmin/papers/2012/2012-bitsch-IPIN-vegemite.pdf
Peter, M., Jafri, S. R. U. N., & Vosselman, G. (2017). Line Segmentation of 2D Laser Scanner Point Clouds for Indoor SLAM based on A Range of Residuals. In *the 3rd ISPRS Workshop on Indoor3D*. Wuhan, China.
Pham, T. T., Reid, I., Latif, Y., & Gould, S. (2015). Hierarchical higher-order regression forest fields: An application to 3D indoor scene labelling. In *Proceedings of the IEEE International Conference on Computer Vision* (Vol. 2015 Inter, pp. 2246–2254). https://doi.org/10.1109/ICCV.2015.259
Qi, C. R., Su, H., Mo, K., & Guibas, L. J. (2016). PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. *Neural Networks (IJCNN), 2016 International Joint Conference on*, 1578–1584. https://doi.org/10.1109/CVPR.2017.16
Quintana, B., Prieto, S. A., Adán, A., & Bosché, F. (2016). Door detection in 3D colored laser scans for autonomous indoor navigation. In *2016 International Conference on Indoor Positioning and Indoor Navigation, IPIN 2016* (pp. 4–7). https://doi.org/10.1109/IPIN.2016.7743677
Rottmann, A., Mozos, Ó. M., Stachniss, C., & Burgard, W. (2005). Semantic place classification of indoor environments with mobile robots using boosting. In *Proceedings of the Twentieth National Conference on Artificial Intelligence, and the Seventeenth Annual Conference on Innovative Applications of Artificial Intelligence and the Seventeenth Annual Conference on Innovative Applications of Artificial Intelli* (pp. 1306–1311). Retrieved from http://www2.informatik.uni-freiburg.de/~stachnis/pdf/rottmann05aaai.pdf
Sahin, C. (2015). Planar segmentation of indoor terrestrial laser scanning point clouds via distance function from a point to a plane. *Optics and Lasers in Engineering*, *64*, 23–31. https://doi.org/10.1016/j.optlaseng.2014.07.007
Schnabel, R., Wahl, R., & Klein, R. (2007). Efficient RANSAC for Point-Cloud Shape Detection, *26*(2), 214–226.
Shi, L., Kodagoda, S., & Ranasinghe, R. (2011). Fast Indoor Scene Classification Using 3D Point Clouds.

In *Conference on Robotics* … (pp. 7–9). EuMA.
Shui, W., Liu, J., Ren, P., Maddock, S., & Zhou, M. (2016). Automatic planar shape segmentation from indoor point clouds. In *Proceedings of the 15th ACM SIGGRAPH Conference on Virtual-Reality Continuum and Its Applications in Industry - VRCAI '16* (pp. 363–372). New York, New York, USA: ACM Press. https://doi.org/10.1145/3013971.3014008
Siadat, A., Kaske, A., Klausmann, S., Dufaut, M., & Husson, R. (1997). An Optimized Segmentation Method for a 2D Laser-Scanner Applied to Mobile Robot Navigation. *IFAC Proceedings Volumes*, *30*(7), 149–154. https://doi.org/10.1016/S1474-6670(17)43255-1
Susperregi, L., Martínez-Otzeta, J. M., Ansuategui, A., Ibarguren, A., & Sierra, B. (2013). RGB-D, laser and thermal sensor fusion for people following in a mobile robot. *International Journal of Advanced Robotic Systems*, *10*. https://doi.org/10.5772/56123
Tschannen, M., Cavigelli, L., Mentzer, F., Wiatowski, T., & Benini, L. (2016). Deep Structured Features for Semantic Segmentation. Retrieved from http://arxiv.org/abs/1609.07916
Tuttas, S., & Stilla, U. (2009). Window detection in sparse point clouds using indoor points. *Information Sciences*, *38*, 131–136. https://doi.org/10.5194/isprsarchives-XXXVIII-3-W22-131-2011
Vosselman, G., & Maas, H.-G. (2010). *Airborne and terrestrial laser scanning*. Dunbeath: Whittles Publishing. Retrieved from http://ezproxy.utwente.nl:2200/patron/FullRecord.aspxp=3417283
Vosselman, G., Coenen, M., & Rottensteiner, F. (2017). Contextual segment-based classification of airborne laser scanner data. *ISPRS Journal of Photogrammetry and Remote Sensing*, *128*, 354–371. https://doi.org/10.1016/j.isprsjprs.2017.03.010
Wen, C., Qin, L., Zhu, Q., Wang, C., & Li, J. (2014). Three-dimensional indoor mobile mapping with fusion of two-dimensional laser scanner and RGB-D camera data. *IEEE Geoscience and Remote Sensing Letters*, *11*(4), 843–847. https://doi.org/10.1109/LGRS.2013.2279872
Xu, Z., Wu, L., Shen, Y., Li, F., Wang, Q., & Wang, R. (2014). Tridimensional reconstruction applied to cultural heritage with the use of camera-equipped UAV and terrestrial laser scanner. *Remote Sensing*, *6*(11), 10413–10434. https://doi.org/10.3390/rs61110413
Zancajo-Blazquez, S., Laguela-Lopez, S., Gonzalez-Aguilera, D., & Martinez-Sanchez, J. (2015). Segmentation of indoor mapping point clouds applied to crime scenes reconstruction. *IEEE Transactions on Information Forensics and Security*, *10*(7), 1350–1358. https://doi.org/10.1109/TIFS.2015.2407699