Closed Door Detection via Point Cloud Data

by

151220172030 Heval Can Aslan ÖZEN

151220162030 Ömer Faruk KARABOSTAN

A Graduation Project Report

Electrical Electronics Engineering Department

JANUARY 2022

Closed Door Detection via Point Cloud Data

by

151220172030 Heval Can Aslan ÖZEN
151220162030 Ömer Faruk KARABOSTAN

A Report Presented in Partial Fulfilment of the Requirements for the Degree Bachelor of Science in Electrical Electronics Engineering

ESKISEHIR OSMANGAZI UNIVERSITY

JANUARY 2022

Closed Door Detection via Point Cloud Data

by

151220172030 Heval Can Aslan ÖZEN

151220162030 Ömer Faruk KARABOSTAN

has been approved by

Supervisory Committee

Asst. Prof. Dr. Burak KALECİ

Asst. Prof. Dr. Helin DUTAĞACI

Asst. Muhammed Oğuz TAŞ

ABSTRACT

The closed-door detection problem has been one of the favored research areas in different domains such as robotics and building information modeling. The main reason of that is door locations separate important parts of the environment such as rooms and corridors. Previous studies that focused on closed-door detection problem generally employed visual data. However, the success of those studies depends on the lighting conditions of the environments. In addition, the distance and angle of the door according to the camera position quite affect the performance of the studies. In this study, we utilized point cloud data to overcome these drawbacks of visual data as well as exploit the ability to describe the 3D characteristic of scenes of the point cloud data. We proposed a rule-based approach to identify closed doors and determine door positions. The rules were extracted regarding the relationship between walls and hinged doors. The experiments were conducted with the ESOGU DOORS dataset to analyze the effectiveness of the proposed method. The test results showed that door detection rate was 95.93%.
Keywords: *door detection, point cloud data*

ÖZET

Kapalı kapı algılama problemi, robotik ve bina bilgi modellemesi gibi farklı alanlarda tercih edilen araştırma alanlarından biri olmuştur. Bunun temel nedeni, kapı konumlarının oda ve koridor gibi ortamın önemli kısımlarını birbirinden ayırmasıdır. Kapalı kapı algılama problemine odaklanan önceki çalışmalarda genellikle görsel veriler kullanılmıştır. Ancak bu çalışmaların başarısı ortamların aydınlatma koşullarına bağlıdır. Ayrıca kamera konumuna göre kapının mesafesi ve açısı da çalışmaların performansını oldukça etkilemektedir. Bu çalışmada, görsel verilerin bu dezavantajlarının üstesinden gelmek ve nokta bulutu verilerinin sahnelerinin 3B özelliklerini tanımlama yeteneğinden yararlanmak için nokta bulutu verilerini kullandık. Kapalı kapıları tanımlamak ve kapı konumlarını belirlemek için kural tabanlı bir yaklaşım önerdik. Kurallar duvarlar ve menteşeli kapılar arasındaki ilişkiler üzerinden çıkarıldı. Önerilen yöntemin etkinliğini analiz etmek için ESOGÜ DOORS veri seti ile deneyler yapılmıştır. Test sonuçları, kapı algılama oranının %95,93 olduğunu göstermiştir.

Anahtar Kelimeler: *kapı bulma, nokta bulutu verisi*

ACKNOWLEDGEMENT

We would like to thank our supervisor, Asst. Prof. Dr. BURAK KALECİ, for his useful comments and on-going support. Secondly, we would also like to thank our co-adviser Res. Asst. KAYA TURGUT for his guidance.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS AND ABBREVIATIONS

Abbreviation	**Explanation**
RANSAC:	Random Sample Consensus
NOS:	Number of Samples
RBCDD:	Rule Based Closed Door Detection

1. INTRODUCTION

Detection of closed doors is an important issue in the fields of robotics, architecture, and facilitating human life [1]. The main reason for this is that doors divide indoor environments into different sections and play an important role in awareness of accessible areas. For example, being aware of a passage or entry point improves the robots' autonomous mobility [2]. It also lays the foundation for projects based on this awareness, such as the implementation of building information modeling (BIM) [3]. As can be seen in Figure 1, the building modeling has been extracted in detail with the point cloud. [4].

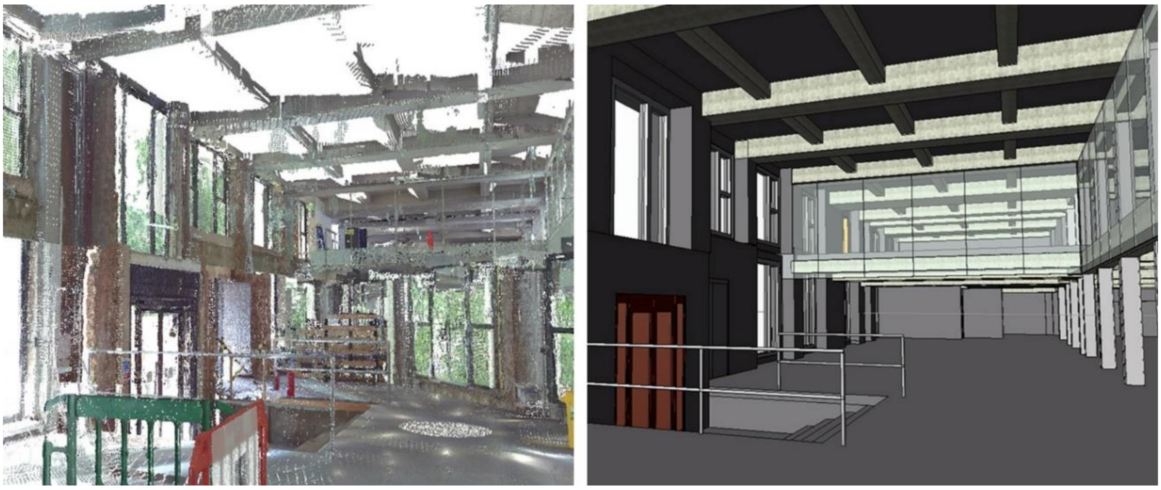


Figure 1: Point Cloud scanning for BIM [4].

The previous studies on the problem of closed-door detection usually aims at identifying closed doors through visual data [5]. However, there are many critical factors affecting the success of these methods. Firstly, the distance and angle between the robot and the door frame may cause lost pixels and incorrectly received visual data. Also, the lighting conditions of the environments can vary the pixels of the visual data, which seriously affect the solutions [6]. On the other hand, some approaches in previous studies used point cloud data to detect closed doors [7]. These studies are not affected by changing light conditions, unlike studies using visual data [8]. However, the angle between the robot and the door frame can affect the success of these methods. Generally, as the angle becomes narrower, the door detection rate decreases [9].

Borgsen et al. [10] utilized point cloud data to find closed doors. They reduced the density of the data containing more than 300,000 points by using voxel-grid-filter to make the process more faster. Then, they separated the door plane from the other data by performing plane segmentation. A perfect plane model was produced using the RANSAC [11] algorithm. In this obtained plane, the door handle or doorknob was searched to detect the door. However, it was observed that the door handle or doorknob was made of reflective metals, thus misleading the sensor. For this reason, the door was tried to be found by searching the keyhole in the door plane. As a result of their tests, they achieved a success of up to 90%. However, they observed that the rate of finding doors dropped rapidly when the distance was more than 2 meters.

Nagahama et al. [12] aimed to detect closed doors using point normals in their work with point cloud data. First, the points were clustered with respect to the normal, and then the planes perpendicular to the ground planes were extracted. Then the door edges were determined, finally the algorithm decides whether there is a closed door or not.

Vilariño et al. [13] used Terrestrial Laser Scanners, which have higher geometric accuracy and can operate at longer distances according to RGB-D cameras. Also, they were able to capture both the point cloud and the corresponding images. They rotated the obtained point cloud so that the walls are parallel to the x or y-axis. Then, the data is segmented using the region growing method. Orthoimage was obtained by using ray-tracing method. Generalized Hough Transform is applied to orthoimages. It uses the Generalized Hough Transform to find similar patterns in an image. Since the doors are rectangular, it is necessary to search for rectangles in the data. Those with the same size and shape as the door cause the wrong decision. In cases where the door and the wall are on the same plane, the accuracy rate is one hundred percent, otherwise, it can be decreased to 33 percent.

Yuan et al. [14] used a RGB-D camera and they obtained a depth image with 320x240 resolution. In this study, it is assumed that the angle between the RGB-D camera and the door is always 90 degrees, so the door is located in the middle of the depth image. It is also assumed that the dimensions of the door are constant 0.9mx2m. By processing the depth image, the furthest points were determined, and the width and height of the plane formed by these points were examined. However, due to the low resolution and the fact that the RGB-D camera can work up to 3.3m, the door finding rates were low.

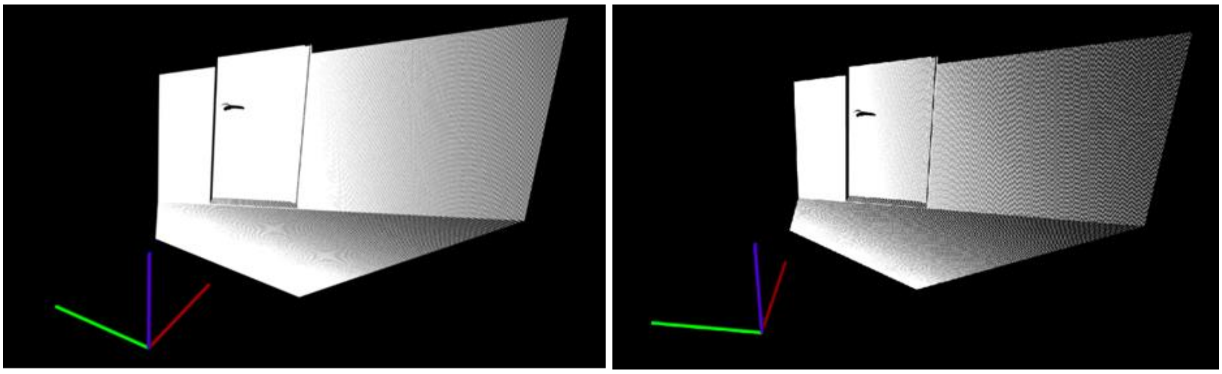
In this study, we present a rule-based closed-door detection (RBCDD) method, which accepts point cloud data as an input. Then, the data is aligned according to heading angle of the robot and the plane segmentation is applied to the rotated data. After that point, plane equations and extreme points of planes are determined. Lastly, a set of rules are defined to detect the closed doors. The experiments were conducted using the ESOGU DOORS dataset and the test result indicates that RBDCC method can identify door with 95% accuracy.

2. METHODOLOG

In this study, RGB-D camera is used to generate point cloud data that has an organized structure with a width and height of 480x640. The point cloud data consists of 307200 points, and each point has x, y, and z coordinates. In the proposed method, the point cloud data is employed to find a closed-door. We follow five basic phases to decide whether a closed-door exists in the scene or not. These basic steps will be explained in detail in this section.

2.1 Aligning point cloud data

The dimensions and the appearance of a closed-door can vary when the point cloud data is captured from different angles and distances according to door frame position. In the proposed method, we first align the point cloud data to the y-z plane to overcome that drawback by rotating it around the z-axis regarding to the robot heading angle. An example of this operation is given in Figure 3. In the figure, the reference coordinate system is shown with red (x-axis), green (y-axis), and blue (z-axis) colors. The raw and rotated point cloud data are depicted in Figure 2(a) and Figure 2(b), respectively. As seen from the figures, the point cloud data is aligned to the y-z plane after rotation.



(a) *(b)*

Figure 2: Aligning point cloud data. (a) Raw point cloud data, (b) Rotated point cloud data.

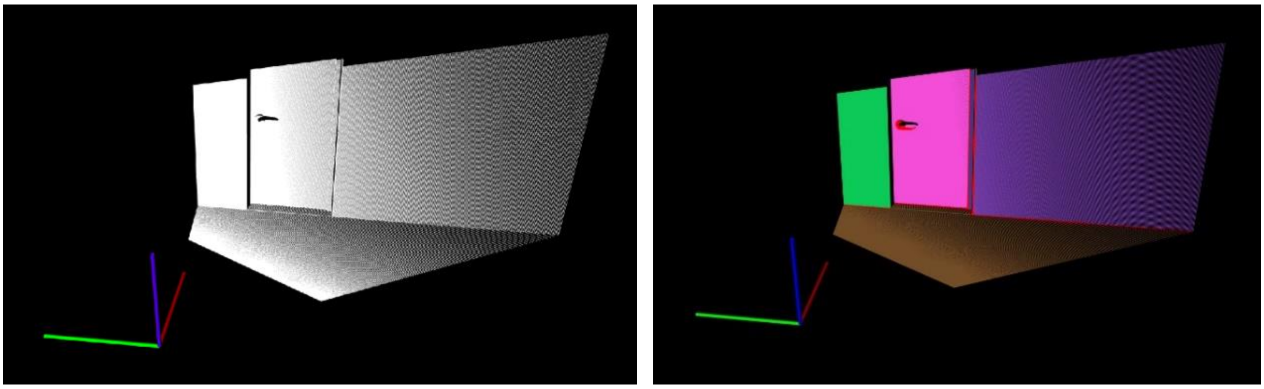
2.2 Plane Segmentation

After the alignment operation is performed, the planes such as the floor, walls, and door frame must be segmented in the second phase since our rule-based closed-door identification approach relies on the relationship between walls and hinged doors. For this purpose, the region growing segmentation [15] method is applied to the rotated point cloud data.

The region-based approaches cluster the points into regions based on local knowledge. Features such as surface orientation, curvature, normal, and others are studied for points within a certain radius or a particular number of neighbors (NumberOfNeighbours) to select the points to be added to a region. Nguyen and Le [16] and Grilli et al. [17] separate region-based approaches into two categories in their reviews: Methods that are top-down (unseeded) and bottom-up (seeded). Top-down approaches begin with a single region that contains all of the points in the point cloud data. They then divide the region into subregions based on a set of criteria. The bottom-up approaches, on the other hand, select some seed points first, and subsequently points that satisfy a specific criterion are added to the corresponding seed point to form a region [18]. Selection of seed locations and the merging criterion are crucial to the success of these approaches [19]. Furthermore, these methods are sensitive to incorrect normal and curvature values, which can result in incorrect results near the region's edge when normal and curvature values change rapidly.

The search radius, or the number of neighbors surrounding of a point, can be raised to avoid these inaccurate results [20]. The elapsed segmentation time will rise in this situation. As a result, with region-based techniques, there is a trade-off between segmentation success and segmenting time [16].

The region-based methods are separated into two categories, the researchers generally prefer to use bottom-up (seeded) methods. Besl and Jain [18] introduced the first seeded region growing algorithm. The algorithm consists of two steps: 1) Determining the seed points, 2) growing the seed to form regions. However, seed selection may vary for other methods. For example, the method used in this study, to select a seed point, Rabbani et al. [21] and Ning et al. [22] first determined an appropriate model for a point and its neighbors. Then, as a seed point, the point that has the smallest distance with the plane created with the neighbor points is chosen. Once the seed point is specified, a search list is generated as pushing the seed point and its neighbors. At that point, the search begins to examine the similarities between the seed point and other list items in terms of local features such as curvature, surface orientation, smoothness, normal, etc.



(a) *(b)*

Figure 3: Plane segmentation. (a) Rotated point cloud data, (b) Point cloud data after plane segmentation.

As we can see in Figure 3 (a) and Figure 3 (b), with the help of the mentioned method, we are able to detect planes and also eliminate unwanted small surfaces indicated by red areas in Figure 3 (b).

2.3 Determining Plane Equation

In the third phase of the proposed method, we must determine the plane equation of segmented planes to define the rules of the closed-door detection approach. The RANSAC is employed to each segmented plane to calculate the plane coefficients. Random sample consensus (RANSAC) is an iterative method for estimating the parameters of a model from a set of observed data [23]. The algorithm starts with the selection of the mathematical model. Then, a small set of points is selected randomly instead of searching the large set of points. The small set is enlarged regarding the DISTANCE_THRESHOLD parameter. The distance between a point and the model is calculated. We assume that the model is plane. The distance value can be considered as the error. If the error is less than the threshold, the point is added to that model and the model is updated. After a model is entirely segmented, the points that belong to the model are extracted from the point cloud and the process is repeated until the number of remaining points reaches the predefined number [20].

2.4 Bounding Box

The RBCDD method needs to know extreme points of the segmented planes as well as plane equations since the door frame must fulfill requirements for the height (DOOR_HEIGHT) and width (DOOR_WIDTH) of the door plane. For that reason, in the fourth phase of our method, we considered bounding box methods [24] such as Axis-Aligned Bounding Box (AABB), Oriented Bounding Box (OBB). Since the bounding box of a set of points is defined as the smallest or minimum enclosed box that all points lay inside the box, calculating the bounding box of a segmented plane can reveal the extreme points of that plane. However, in some cases, a plane can overlap another plane due to the angle and distance between the camera and the door locations. We also assessed the convex hull concept to handle these situations.

2.5 Decision

In the last phase of the proposed method, we gave the assumptions and rules of the method. First, we assumed that a hinged closed-door must place on the wall plane and it must separate the wall plane into two parts. Besides, the door plane must be located behind the wall planes according to the door frame dimensions. Lastly, a door plane must meet heigth and width requirements.

The rules of the methods were extracted depending on these assumptions and given below:

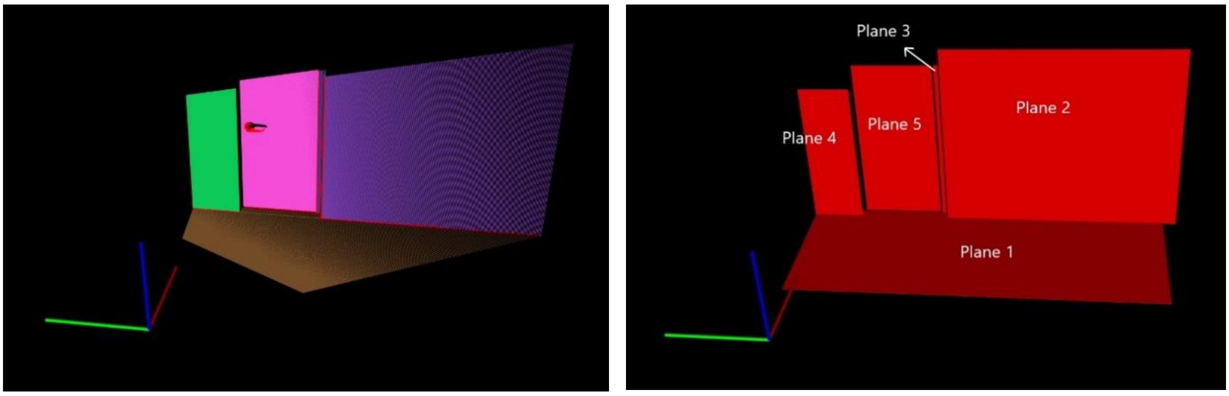
1) The wall and door planes must be parallel, and they must lie on the y-z plane.

2) The door plane must be between two wall planes.

3) There must be a depth difference according to the door frame between the two wall planes and the door plane.

4) Dimensions of the door plane must meet predefined requirements.

An example of the decision phase of the method is given in Figure 4(b). In that example, after point cloud alignment and plane segmentation phases, we have five planes, and their plane equations and extreme points were also stored in an appropriate manner.



(a) *(b)*

Figure 4: Printing bounding boxes. (a) Point cloud data after plane segmentation, (b)Point cloud with bounding boxes

As seen from the figure, planes 2 and 4 describe the wall planes, plane 5 depicts the door plane whereas planes 3 show the door frame, and plane 1 indicates the floor plane. In the decision phase, we first discarded the planes that are not lying on the y-z plane with the aid of plane equations. In the example, planes 3 and 1 were removed. Then, we searched three planes that are parallel to the y-z plane in the remaining planes (planes 2, 4, and 5). After that point, the extreme points of these planes were considered, and the plane located between two other planes was determined (plane 5). If that plane is positioned behind the wall planes (planes 2 and 4), the plane was marked as a candidate door plane. Lastly, candidate door planes were evaluated for dimension requirements. If the difference between predetermined and candidate's width and height values is less than a threshold, the candidate was labeled as a closed-door location.

3. EXPERIMENTS

The OGUROB DOORS Dataset [8] was used to analyze the performance of the RBCDD method. Although the OGUROB DOORS Dataset consists of samples for open, half-open, and closed doors, closed-door samples were selected for experiments. In the closed-door dataset, there are 1353 samples with varying angles and distances. Figure 5 shows the positions of the samples according to the door position. The distance between samples is 10 cm of x and y dimensions and the heading angle of the robot is determined randomly. The samples lie between two semi-circles with radii 1.5 and 3.5 meters. Besides, five regions, orange (0º-30º), red (30º-60º), green (60º-120º), yellow (120º-150º), and blue (150º-180º), are constructed depending on the angle between samples and the door frame. The number of samples in each region regarding the distances is given in Table 1.

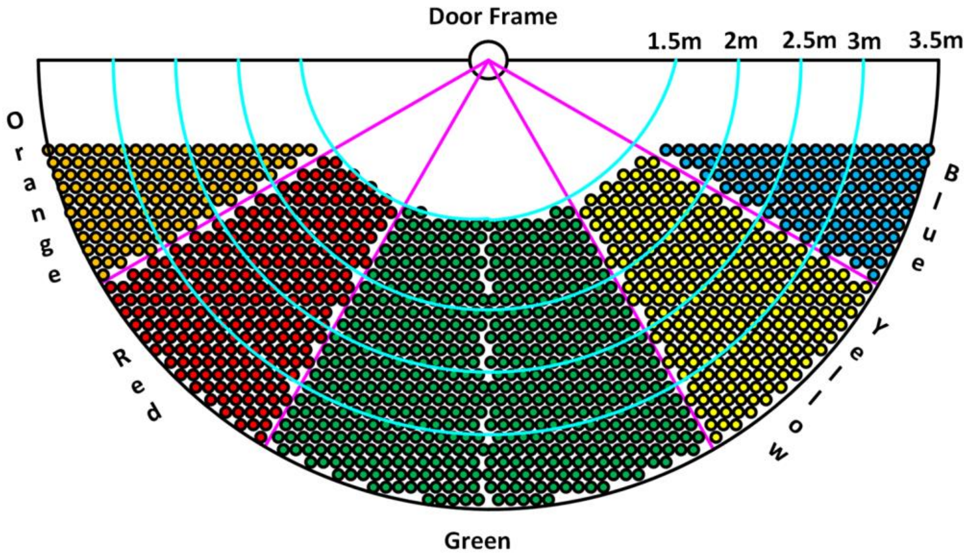


Figure 5: OGUROB DOORS dataset. Regions according to the angle between the robot and the door frame [8].

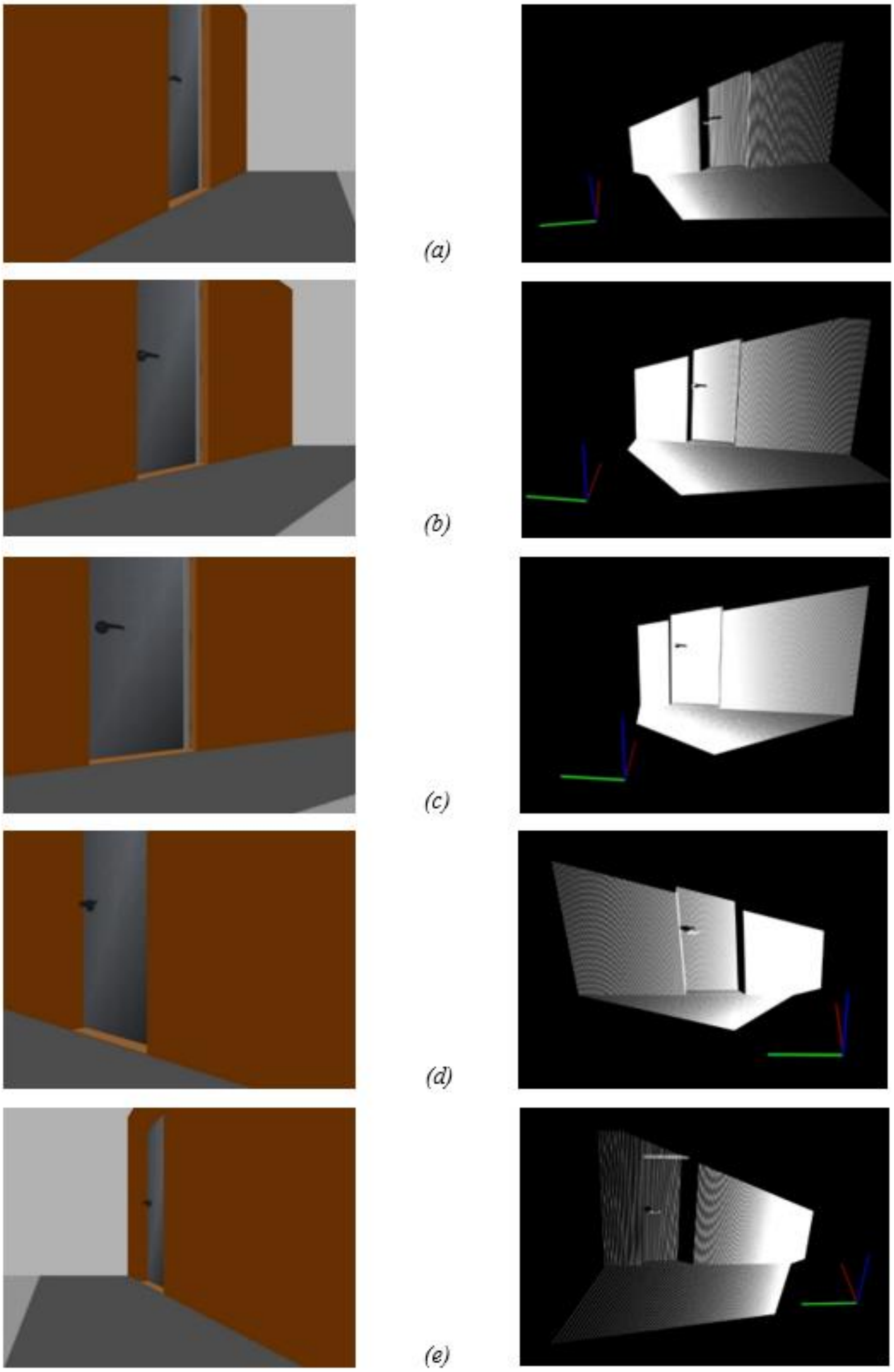
Table 1: Number Of Samples Depending On Angle And Distance

Distance / Angle	1.5-2m	2-2.5m	2.5-3m	3-3.5m	Total
Orange	9	21	41	59	130
Red	50	55	70	91	266
Green	117	115	140	189	561
Yellow	50	55	70	91	266
Blue	9	21	41	59	130
Total	235	267	362	489	1353

Examples for orange, red, green, yellow, and blue regions are given in Figure 6. In the figure, the left column shows the RGB image of the scenes whereas the right column indicates the corresponding point clouds.

The RBCDD method was implemented with C++ programming language using Point Cloud Library (PCL) [7]. In the first phase, matrix transform [25] in the PCL was used to align the point cloud data to y-z plane. The angle parameter of the matrix transform was the heading angle of the robot. Then, region growing method [15] was employed to plane segmentation. The NumberOfNeigbours, SmoothnessThreshold, and CurvatureThreshold parameters of region growing was selected as 30, 3 and 1, respectively. To determine the plane coefficient RANSAC [11] algorithm was performed. DistanceThreshold parameter of RANSAC was chosen as 0.01. At that point, PCL bounding box method [24] was used to obtain bounding boxes of the planes. Lastly, considering the door standards in Turkey [26], DOOR_HEIGHT parameter was settled as 2 meters and the DOOR_WIDTH parameter as 0.9 meters.

In order to conduct experiments for RBCDD, test sets were created by randomly selecting 26 samples from the orange region, 53 from the red region, 113 from the green region, 53 from the yellow region and 26 from the blue region. The experiments were conducted on a PC with the AMD Ryzen™ 5 1500X Processor 16GB RAM, and Ubuntu 20.04 operating system.



(a)

(b)

(c)

(d)

(e)

Figure 6: (a) Orange Area, (b) Red Area, (c) Green Area, (d) Yellow Area, (e) Blue Area

In order to examine the success of the RBCDD method, accuracy was calculated as follows:

$$\text{Accuracy} = \frac{\text{number of correctly predicted samples}}{\text{total number of samples}} \tag{1}$$

As can be seen in Table 2, the success rate of the method is over 98% outside the green zone. In the orange, red and yellow region, there are examples that fall outside the set of rules. The majority of the errors in these regions were observed as the mismatch of the width and height parameters or the wall or door thickness exceeding the tolerable level as a result of the bounding box method.

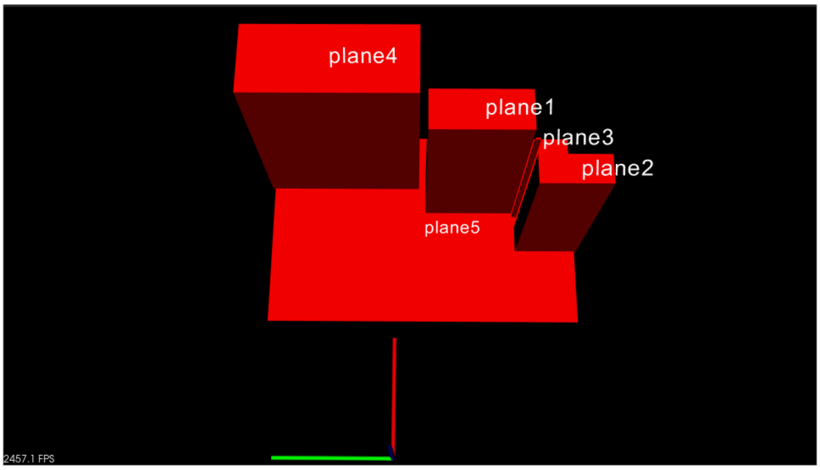


Figure 7: Bounding Box Error.

As we said before, point cloud data is used after it is rotated around the z axis according to the angle obtained from the robot. Thus, the walls become parallel to the y-z plane. However, in some cases, it has been determined that this angle does not rotate the data enough, and the bounding boxes are different from the desired one due to the use of the Axis-Aligned Bounding Box.

As can be seen in Figure 7, while we expected the bounding boxes to be almost 2-dimensional as in Figure 4(b), a different bounding box than expected appeared. This situation affects the decision mechanism and causes wrong decision making.

Table 2: Test Results

Angle	Distance	NOS	RBCDD
0-30	1.5-2	2	%100
	2-2.5	4	%100
	2.5-3	8	%97.5
	3-3.5	12	%100
30-60	1.5-2	10	%100
	2-2.5	11	%98.18
	2.5-3	14	%100
	3-3.5	18	%100
60-120	1.5-2	24	%99.17
	2-2.5	23	%94.78
	2.5-3	28	%73.57
	3-3.5	38	%97.37
120-150	1.5-2	10	%100
	2-2.5	11	%98.18
	2.5-3	14	%100
	3-3.5	18	%97.78
150-180	1.5-2	2	%100
	2-2.5	4	%100
	2.5-3	8	%100
	3-3.5	12	%100
			%95.93

4.PROJECT PLAN

4.1 Work Packages

- **Work Package 1 – Examining the language and applications to be used.**
- **Work Package 2 – Examination of the methods to be used in the search process.**
- **Work Package 3 – Examination of segmentation processes.**
- **Work Package 4 – Region Growing Segmentation implementation.**
- **Work Package 5 – Determining thresholds for Segmentation and testing.**
- **Work Package 6 – Cluster extraction and writing clusters to file.**
- **Work Package 7 – RANSAC implementation.**
- **Work Package 8 – Transformation implementation.**
- **Work Package 9 – Convex Hull and Concave Hull implementation:**

- **Work Package 10 – Bounding box implementation.**
- **Work Package 11 – Editing code for testing and testing data.**

Table 3: Work Packages

Work Package		Starting date	Ending date	Assigned to
Examining the language and applications to be used.	1	26.03.2021	14.10.2021	H.C.A. ÖZEN Ö.F. KARABOSTAN
Examination of the methods to be used in the search process.	2	14.10.2021	21.10.2021	H.C.A. ÖZEN Ö.F. KARABOSTAN
Examination of segmentation processes	3	21.10.2021	28.10.2021	H.C.A. ÖZEN Ö.F. KARABOSTAN
Region Growing Segmentation implementation.	4	28.10.2021	04.11.2021	H.C.A. ÖZEN Ö.F. KARABOSTAN
Determining thresholds for Segmentation and testing.	5	04.11.2021	11.11.2021	H.C.A. ÖZEN
Cluster extraction and writing clusters to file.	6	11.11.2021	17.11.2021	H.C.A. ÖZEN
RANSAC implementation.	7	17.11.2021	02.12.2021	H.C.A. ÖZEN Ö.F. KARABOSTAN
Transformation implementation.	8	02.12.2021	09.12.2021	H.C.A. ÖZEN Ö.F. KARABOSTAN
Convex Hull and Concave Hull implementation	9	09.12.2021	16.12.2021	Ö.F. KARABOSTAN
Bounding box implementation	10	16.12.2021	23.12.2021	Ö.F. KARABOSTAN
Editing code for testing and testing data	11	23.12.2021	30.12.2021	H.C.A. ÖZEN Ö.F. KARABOSTAN

5. CONCLUSION

In this study, we aimed to detect closed doors using a rule-based algorithm and point cloud data. The success of existing studies depends on the distance and angle between the door frame and the sensor. Therefore, we tried to implement an angle-invariant algorithm. Test results show that RBCDD can classify doors with over 95% overall success. Increasing accuracy of the RBCDD method is planned for future works. This can be achieved by determining a new rule for the region with the most mistakes or filtering it to make it compatible with the existing rules.

REFERENCES

[1] S. A. Prieto, A. Adán, A. S. Vázquez, and B. Quintana, “Passing through open/closed doors: A solution for 3D scanning robots,” Sensors, vol. 19, no. 21, p. 4740, 2019.

[2] B. Kakillioglu, K. Ozcan and S. Velipasalar, "Doorway detection for autonomous indoor navigation of unmanned vehicles," 2016 IEEE International Conference on Image Processing (ICIP), 2016, pp. 3837-3841, doi: 10.1109/ICIP.2016.7533078.

[3] J. Jung, C. Stachniss, S. Ju, and J. Heo, “Automated 3D volumetric reconstruction of multiple-room building interiors for AS-built bim,” Advanced Engineering Informatics, vol. 38, pp. 811–825, 2018.

[4] “Point cloud scan to BIM: Point cloud modeling services,” *Point Cloud Scan to BIM | Point Cloud Modeling Services*. [Online]. Available: https://www.pointcloudbimmodeling.com/. [Accessed: 13-Jan-2022].

[5] U. G. Adar and L. Bayindir, “Door detection using camera images obtained from indoor enviroments,” 2015 23nd Signal Processing and Communications Applications Conference (SIU), 2015.

[6] X. Yang and Y. Tian, “Robust door detection in unfamiliar environments by combining edge and corner features,” 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops, 2010.

[7] R. B. Rusu and S. Cousins, “3D is here: Point cloud library (PCL),” 2011 IEEE International Conference on Robotics and Automation, 2011.

[8] B. Kaleci and K. Turgut , "COMPARISON OF DEEP LEARNING TECHNIQUES FOR DETECTION OF DOORS IN INDOOR ENVIRONMENTS", Eskişehir Osmangazi Üniversitesi Mühendislik ve Mimarlık Fakültesi Dergisi, vol. 29, no. 3, pp. 396-412, Dec. 2021, doi:10.31796/ogummf.889095

[9] L. Souto, A. Castro, L. Gonçalves, and T. Nascimento, “Stairs and Doors Recognition as Natural Landmarks Based on Clouds of 3D Edge-Points from RGB-D Sensors for Mobile Robot Localization,” Sensors, vol. 17, no. 8, p. 1824, Aug. 2017.

[10] Borgsen, S. M. Z., Schöpfer, M., Ziegler, L. & Wachsmuth, S. (2014). Automated door detection with a 3Dsensor. Canadian Conference on Computer and Robot Vision, 276-282, Montreal, QC, Canada

[11] *Point Cloud Library (PCL): Pcl::randomsampleconsensus< pointt > class template reference*. [Online]. Available: https://pointclouds.org/documentation/classpcl_1_1_random_sample_consensus.html . [Accessed: 18-Jan-2022].

[12] K. Nagahama, K. Takeshita, H. Yaguchi, K. Yamazaki, T. Yamamoto, and M. Inaba, “Estimating door shape and manipulation model for daily assistive robots based on the integration of visual and touch information,” *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018.

[13] Díaz-Vilariño, L.; Khoshelham, K.; Martínez-Sánchez, J.; Arias, P. 3D Modeling of Building Indoor Spaces and Closed Doors from Imagery and Point Clouds. *Sensors* 2015, *15*, 3491-3512.

[14] Yuan, T. H., Hashim, F. H., Zaki W. M. D. W. & Huddin, A. B. (2015). An automated 3D scanning algorithm using depth cameras for door detection. International Electronics Symposium (IES), 58-61, Surabaya, Indonesia.

[15] “Region growing segmentation,” *Region growing segmentation - Point Cloud Library 0.0 documentation*. [Online]. Available: https://pcl.readthedocs.io/projects/tutorials/en/latest/region_growing_segmentation.html#region-growing-segmentation. [Accessed: 13-Jan-2022].

[16] A. Nguyen and B. Le, “3D point cloud segmentation: A survey,” *2013 6th IEEE Conference on Robotics, Automation and Mechatronics (RAM)*, 2013.

[17] E. Grilli, F. Menna, and F. Remondino, “A review ofpoint clouds segmentation and classification algorithms,” *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XLII-2/W3, pp. 339–344, 2017.

[18] P. J. Besl and R. C. Jain, “Segmentation through variable-order surface fitting,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 10, no. 2, pp. 167–192, 1988.

[19] Y. Xie, J. Tian, and X. X. Zhu, “Linking points with labels in 3D: A Review of Point Cloud Semantic Segmentation,” *IEEE Geoscience and Remote Sensing Magazine*, vol. 8, no. 4, pp. 38–59, 2020.

[20] E. E. ERUYAR, M. YILMAZ, B. YILMAZ, O. AKBULUT, K. TURGUT, and B. KALECİ, “A comparative study for indoor planar surface segmentation via 3D Laser Point Cloud Data,” *Black Sea Journal of Engineering and Science*, 2020.

[21] Rabbani T, Van Den Heuvel F, Vosselmann G. 2006. Segmentation of point clouds using smoothness constraint. Int archives of Photogrammetry, Remote Sens and spatial Inf Sci, 36(5): 248-253

[22] Ning X, Zhang X, Wang Y, Jaeger M. 2009. Segmentation of architecture shape information from 3D point cloud. Proceedings of the 8th International Conference on Virtual Reality Continuum and its Applications in Industry, 127-132.

[23] M. A. Fischler and O. Firschein, in *Readings in computer vision: Issues, problems, principles, and paradigms*, Los Altos (California): Morgan Kaufmann, 1987, pp. 726–740.

[24] *Point Cloud Library (PCL): Pcl::boundingboxxyz struct reference*. [Online]. Available: https://pointclouds.org/documentation/structpcl_1_1_bounding_box_x_y_z.html. [Accessed: 18-Jan-2022].

[25] "Using a matrix to transform a point cloud," *Using a matrix to transform a point cloud - Point Cloud Library 1.12.1-dev documentation*. [Online]. Available: https://pointclouds.org/documentation/tutorials/matrix_transform.html. [Accessed: 26-Jan-2022].

[26] TS 9111- Özürlüler ve hareket kısıtlılığı bulunan kişiler için binalarda ulaşılabilirlik gerekleri, (2020). https://intweb.tse.org.tr/Standard/Standard/Standard.aspx?081118051115108051104119110104055047105102120088111043113104073097107103081101066102086100073109.