Geometric Feature Extraction from 2D Laser Range Data for Mobile Robot Navigation

Tharindu Weerakoon, Kazuo Ishii, Amir Ali Forough Nassiraei
Department of Human Intelligence Systems
Kyushu Institute of Technology
2-4, Hibikino, Wakamatsu, Kitakyushu, 808-0196, Japan
{weerakoon-tharindu@edu, ishii@, nassiraei@}.brain.kyutech.ac.jp

***Abstract*— Feature extraction and segmentation obviously play an utmost important role in autonomous mobile robot localization and navigation. In this paper we discuss some line segmentation and feature extraction algorithms and proposed an adaptive feature extraction algorithm for 2D laser range data. Features in indoor environments that are considered in this paper can be described as two geometric primitives: line segments which represent the walls, and corners. Segmentation process estimates the line segments belong to the walls and which represent the same object have been grouped together. Intersecting points of the line segments, called corners are treated as landmarks. Some segmentation algorithms are implemented and tested using laser range data captured in different environments and the effectiveness of them is analyzed. Finally, a better adaptive feature extraction and segmentation algorithm which generates a line map of an unstructured environment is proposed.**

Keywords—segmentation; landmarks; least square; iterative methods; laser range finder

I. INTRODUCTION

Autonomous mobile robots need to know its own position both in a well-known and in a totally unknown environment. For precise localization, a mobile robot requires an aptitude to acquire the knowledge of the surrounding environment using the information collected by the sensors. This aptitude enables the robot to interact with the environment for making decisions in its motion, in the case of obstacle avoidance, localization, path planning, and mapping. Some self-localization algorithms often use an odometry sensor system for localization which is known as dead reckoning. Odometry or dead reckoning leads to unbounded errors mainly because the error is accumulative over the time.

A number of techniques have been developed to solve this problem and some of them are based on natural landmarks of the surrounding environment. These landmarks are observed over the time and they are used for the localization. One of the fundamental methods to get the environmental information is the use of adaptive sensors such as ultrasonic sensors or laser range finders (LRFs). In terms of accuracy and reliability, LRFs are well suitable sensors that show much better performance compared to the ultrasonic sensors. Therefore, LRFs are commonly employed to extract the features of the environment than the ultrasonic sensors. These features could be corners, vertices or straight lines represent the walls or more complex geometric entities.

However, laser range data consist of random noise and different features because of field of view which makes it difficult to extract the features of the environment. Moreover, if the mobile robot is moving in a complex environment, feature extraction process will be more difficult. Therefore, instead of working with raw data point cloud directly, methods of feature based localization that extract the geometric features of the environment are used.

The work presented here deals with laser range data segmentation, feature extraction and scan-matching which is suitable for self-localization and navigation of mobile robots. In this paper we analyzed some of the existing segmentation and geometric feature extraction techniques and proposed a better segmentation approach. A segment matching and merging algorithm is used to generate a map of the environment and estimate the most precious positions for the landmarks.

II. LINE SEGMENTATION AND FEATURE EXTRACTION

Laser range data segmentation and feature extraction methods can be basically categorized into various approaches: Point-Distance Based Segmentation (PDBS) [1] [2] [3] [4] [5] [6], Kalman Filter Based Segmentation (KFBS), Hough transform method [7], fuzzy clustering based segmentation[3], etc. The PDBS method can be further divided into two categories: constant threshold based methods and Adaptive Breakpoint Detector (ABD) [3] methods.

Segmentation approaches mainly aim to identify and separate set of segments related to the captured data set by the LRF. Feature extraction provides more condensed, feasible and full description for the special features of the objects that can be used as the landmarks in robot navigation.

Data segmentation is the first process performed for the LRF scanned data set. This will give line segments, best fitted line parameters for the segments, non-segmented data points that have to be considered in obstacle avoidance of the mobile robots. Well segmented data set is used to extract the features such as corner points in addition to the line segments. Split-and-merge is one of the most suitable algorithms for line extraction because of its superior speed and capability of real-time localization applications. This method is selected for our

work and the process can be performed in three steps: split, straight line fitting and merge.

A. LRF raw data representation

LRF explores the environment before the line extraction. In one scanning cycle, it acquires N number of data points (P) which can be defined in polar coordinates (α_n, r_n) as shown in Fig. 1(a) and Cartesian coordinates (x_n, y_n) as in (1).

$$P = \left\{p_n = \begin{pmatrix}\alpha_n \\ r_n\end{pmatrix}\right\} = \left\{p_n = \begin{pmatrix}x_n \\ y_n\end{pmatrix}\right\}, n \in [1, N] \quad (1)$$

α_n represents the beam angle and the r_n is the distance to the points on the object in the robot's frame work. Segments S_i of the 2D point cloud can be defined as;

$$S_i = \{(\theta_i, d_i), (P_i^s, P_i^e); | \ i = l{:}m\}, 1 \le l < m < N \quad (2)$$

Where (θ_i, d_i) defines the Hough transformed parameters of the line segments and (P_i^{e1}, P_i^{e2}) represents the end point of segments as shown in Fig.1.

In our discussion we target only the PDBS methods. These methods are based on the Euclidean distance between the points for the break-point detection. In the next sub section we discussed some existing PDBS methods briefly.

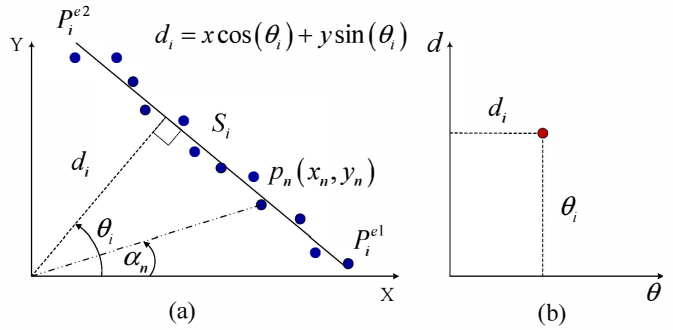


Fig. 1. The principle of Hough transformation,(a) representation of LRF scanned points of a line in $\boldsymbol{xy}$-plane, (b) representation of the line in $\boldsymbol{d\theta}$-plane

B. Polyline splitting of scanned data

This process searches the preliminary end points of all the segments. Splitting of scanned data is kind of data pre-processing which splits the scanned point cloud into set of collinear points. In this work, we consider adaptive breakpoint detection (ADB) method to find point-to-point distance threshold and iterative end point fitting (IEPF) methods for splitting the data points into set of groups. ADB algorithm determines the minimum distance threshold between two consecutive points as described in Fig. 2. Breakpoint detection is a procedure of great importance for line segmentation algorithm because, without such a task, the line extractor is tend to connect two neighbor lines, even if there is a large discontinuity between them. The purpose of ABD is to check if there exists a discontinuity between consecutive laser range scan points, say p_n and p_{n+1}.

Simple breakpoint detector does check the distance between two consecutive points p_{n+1} and p_n, $D(r_n, r_{n+1}) = \| p_{n+1} - p_n \|$ with the distance threshold D_{thd}.

$$If\ D(r_n, r_{n+1}) > D_{thd}\ ,\ \ Break\ Point \quad (3)$$

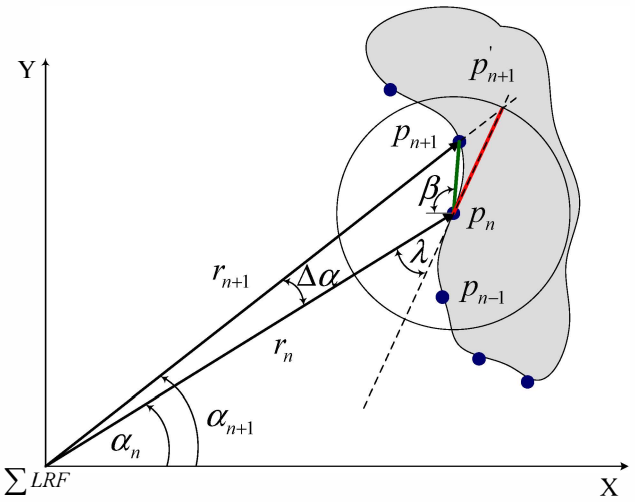


Fig. 2. Principle of break point detection using ABD method

The main difficult task in this process is the computation of the threshold value D_{thd}. Some various threshold calculation approaches have been proposed in literature. The methodology explained in Fig.2 considers the range scan distance r_n in the threshold computation process. In this approach a virtual line passing through the point p_n, which represents the extremum case where an environmental line can be detected reliably has been introduced. The angle λ is the minimum angle that the virtual line inclines with the laser beam direction. Under this constrains, as explained in [3], threshold distance for the point p_n can be calculated as bellow.

$$D_{thd} \cdot \sin(\lambda - \Delta\alpha) = r_n \cdot \sin(\Delta\alpha)$$

$$D_{thd} = \frac{r_n \cdot \sin(\Delta\alpha)}{\sin(\lambda - \Delta\alpha)} \quad (4)$$

To accompany with the sensor noise on the measured data, a parameter σ_r which is the residual variance of the LRF is added to the threshold calculation. Then the threshold can be re-written as;

$$D_{thd} = r_n \cdot \frac{sin(\Delta\alpha)}{sin(\lambda - \Delta\alpha)} + 3\sigma_r \quad (5)$$

Some other techniques to calculate the threshold distance D_{thd} have been proposed. Though the method in (5) shows good performance it fails to give adaptive threshold calculations in some situations. By considering r_n and r_{n+1}, with a little modification of (5), in [8] the utilization of an adaptive threshold D_{thd} is proposed as shown in (6).

$$D_{thd} = min\{r_n, r_{n+1}\} \cdot \frac{sin(\Delta\alpha)}{sin(\lambda - \Delta\alpha)} + 3\sigma_r \quad (6)$$

If the measurement data p_n are considered to iterate in $n = 1 \rightarrow N$ direction, a different result will be obtained than in $n = N \rightarrow 1$.

Another well-known method proposed in [9], the D_{thd} is defined as in (7).

$$D_{thd} = C_0 + C_1 min\{r_n, r_{n+1}\} \quad (7)$$

Where C_0 is a constant parameter and C_1 can be expressed as $C_1 = \sqrt{2(1 - cos\Delta\alpha)} = D(r_n, r_{n+1})/r_n$.

A modified version of this method has been proposed in [10] by including a new parameter β (see Fig. 2) to reduce the dependency of the segmentation. The proposed threshold condition can be expressed as:

$$D_{thd} = C_0 + \frac{C_1 min\{r_n, r_{n+1}\}}{\cot\beta\left[cos\left(\Delta\alpha/_2\right) - sin\left(\Delta\alpha/_2\right)\right]} \quad (8)$$

A simple way to calculate the threshold distance has been proposed in [11].

$$D_{thd} = \left|\frac{r_n - r_{n+1}}{r_n + r_{n+1}}\right| \quad (9)$$

Basic scenario of the breakpoint detection algorithm is shown in Fig.3. For each scanned data point ABD algorithm calculates the threshold distance D_{thd} for the next consecutive point and this value is compared with the distance between the two points. If the distance is greater than the threshold, breakpoint will be detected and the data points will be divided into two clusters as shown in Fig. 3.

After performing the basic breakpoint detection using ABD method, IEPF algorithm was used to split the segments further. This process was performed for each of the basic segments and its principle of operation is shown in Fig. 4. It will first connect the two end points and find the maximum orthogonal distance from the point, if the distance is greater than the given threshold, segment will be split into two at this point.

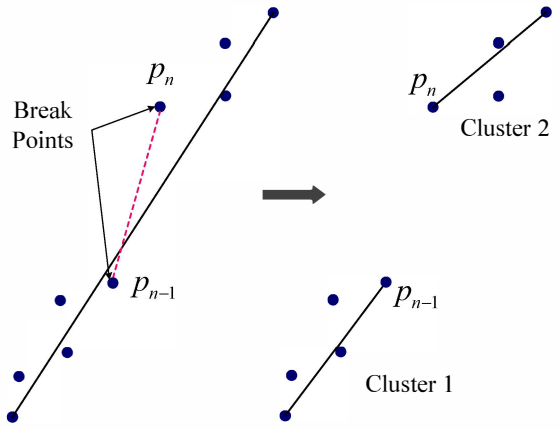


Fig. 3. Principle of breakpoint detection algorithm

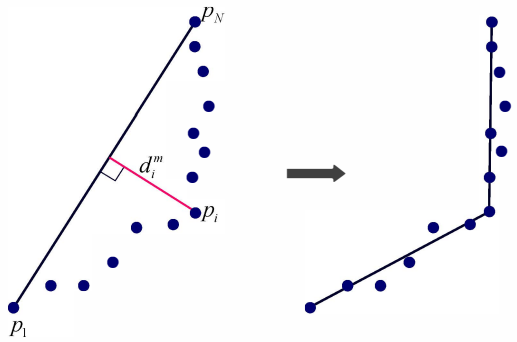


Fig. 4. Principle of IEPF algorithm

C. Straight line fitting

After splitting of the scanned data set, the pre-processing algorithm will give set of data points which belong to different environment features (segments). Each of these data groups are subjected to line fitting algorithm which gives the best fitting line segment for them. Orthogonal least square (OLS) method also known as total least squares method is employed to find the best fitting line segments and corresponding estimation for scanned points. The crossing points of the line segments, also called the corner points are considered as the landmarks.

D. Landmark detection

All the identified end points of the segments by the above process will not be suitable to consider as the landmarks for mobile robot navigation. Landmarks must be selected carefully from these candidate features. To make a corner in an indoor environment, two walls have to meet and hence the intersect points of the segments where the end points are close enough are treated as the landmarks. In the landmark selection algorithm, the corner points which are located very closely will be considered as the same corner point. Moreover, if the line segments' inner intersecting angle is larger than θ_{thd} (= 30º for this study), the corners are considered as the landmarks so that the position error can be reduced.

E. Line segment merging

Line segments that have closely similar parameters are merged into one line and the corresponding scanned points are then re-processed through the OLS algorithm to get the best fitted line. Polar parameters of the line segments $[\theta_i, d_i]$ for K-D tree Nearest-neighbor search (NNS) algorithm are used to check the similarity and find the similar lines. The polar form of the estimated line segment can be represented as in (10).

$$d_i = xcos\theta_i + ysin\theta_i \quad (10)$$

Where $-\pi < \theta \leq \pi$ is the angle between the x axis and d is the perpendicular distance to the line from the origin.

Then we introduced an end-point threshold D^e_{thd} for line segment merging. If the end point distance is less than the threshold D^e_{thd}, corresponding segments will be subjected to merge together. By this process we can eliminate some bad similarity calculation results of NNS method.

In the real application, as the robot is moving it will scan the environment continuously and extract the line segments and landmarks. This segment and landmark data are stored in a database and newly extracted line segments should be merged with the existing data. In this process new data will be matched with the data history and the similar data set which represents the same objects will be extracted using NNS algorithm. Newly scanned line segments can have different end points but closely similar polar parameters with the segment history. Due to the different scanning views, we will have different situations where a new line segment can be considered with the existing ones to merge and extend the line features. Fig. 5 shows four different scenarios for extending the line feature with newly extracted data.

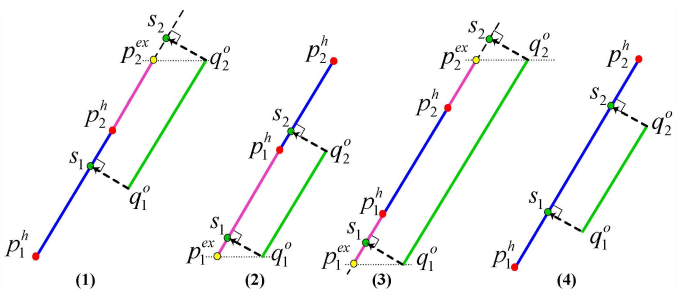


Fig. 5. Different scenarios for extending line segments

The blue color line (p_1^h, p_2^h) represents the matched line segment and (q_1^o, q_2^o) is the newly observed segment while p_2^{ex} is the extended point of p_2^h of the line segment. Fig. 5(1) shows the extension of the point p_2^h, for example, detection of a wall as the robot is moving forward. End point of the segment p_1^h has extended to p_1^{ex} as in the plot 2. This will happen when robot moves backward to a wall in the right side of it. Both the end points of the segment p_1^h and p_2^h have been extended to p_1^{ex} and p_2^{ex} as in the plot 3, and this kind of association arises when the robot moves towards a wall.

In order to find the extending part of the line segments, we considered orthogonal projections of the end points of the new one on that of the existing. By selecting the inner and outer projection points, the end points of the line segments were updated. We kept remain the slope of the existing line segment because it was estimated correctly based on the previous information. The end points to be extended were selected as explained bellow.

$$K^{s_1} = \overrightarrow{P_1^h P_2^h} \cdot \overrightarrow{s_1 P_2^h} \qquad K^{s_2} = \overrightarrow{P_1^h P_2^h} \cdot \overrightarrow{s_2 P_2^h}$$

$$d_h^2 = \left\|(P_1^h, P_2^h)\right\|^2$$

1) Case 1: $K^{s_1} > 0 \quad and \quad K^{s_2} > 0$

$$If \quad K^{s_2} < d_h^2 \ and \quad K^{s_2} > d_h^2$$

The point s_1 is in between the segment end points (P_1^h, P_2^h) and segment has to be extended from P_1^h to s_2 direction or wise versa. If $K^{s_1} \gtrless d_h^2 \ and \ K^{s_2} \gtrless d_h^2$, both the points would be in the outside of the segment but in the same side or in between the end points and then no extension is performed. In case of the selection of s_1 and s_2 is reversed, the above result will not be changed.

2) Case 2: $K^{s_1}.K^{s_2} < 0$

$$If \ K^{s_1} > d_h^2$$

Then the point s_1 and s_2 both are in outside the segment end points. Then the line segment should be extended to both the directions; P_1^h to s_1 and P_2^h to s_2. But if $K^{s_1} < d_h^2$ point s_1 is inside the segment and P_2^h need to be extended to s_2.

3) Case 3: $K^{s_1} < 0 \ \ and \ \ K^{s_2} < 0$

In this situation, both the end points are outside but the same side. Therefore, no extension is considered.

This checking process should be performed for both the points $s_1$and s_2 and to calculate the extended point of the line segment slope of the segment m should be used. Calculation process of coordinates of the extended points is explained bellow.

As shown in the plot 1 of the Fig. 5, the line extension can be calculated from p_2^h to p_2^{ex}. When the slop of the segment $m > 1$ the coordinates of the extended point p_2^{ex} can be calculated as bellow.

$$y_{p_2^{ex}} = y_{p_2^h}$$

$$x_{p_2^{ex}} = f^{-1}\left(y_{p_2^{ex}}\right)$$

If the slope $m < 1$ the line extension can be computed as bellow for the same case of plot 1 of Fig 5.

$$x_{p_2^{ex}} = x_{p_2^h}$$

$$y_{p_2^{ex}} = f\left(x_{p_2^{ex}}\right)$$

III. Experimental Study and Results

Experiment is performed to extract the line segment features and the corner points as landmarks in a structured indoor environment. Layout of the experimental field is shown in Fig. 6 and laser range scanner (Hokuyo URG-04LX) is used to extract the environmental information. All the measured data captured by the LRF is used for the feature extraction algorithm and it is expected to detect some of the visible corner points denoted by C1~C7 while calculating the best fitting line segment parameters for the walls which are detected. LRF is capable of detecting 4.0 m distance with the maximum scanning range of 240º.

Flowchart of the segmentation and feature extraction algorithm is shown in Fig 7. After processing the LRF scanned data through ADB and IEPF algorithms, data points of each line segment was pre-processed before sending through the Best Line Fitting (BLF) algorithm to estimate the line segment features. In this post-processing, the effect of the corner data points of the consecutive segment could be eliminated.

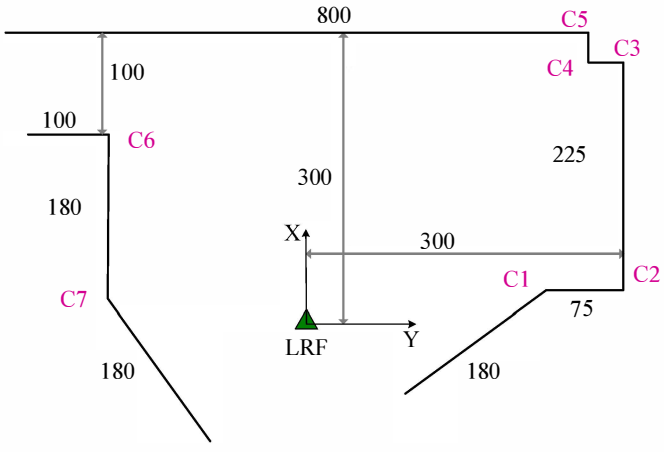


Fig. 6. Floor layout of the experimental world(in cm)

Fig. 8 shows the map of an indoor environment generated using the raw LRF data for a single scan. This data was used to estimate the line segment parameters which represent the walls

and the corner points which are called landmarks. Estimated line segments and corner landmark points are shown in Fig 9, and the true coordinates and the estimated coordinates of the corner points are given in the Table I. It clearly shows that the algorithm shows expected performance of estimating the corner landmark points and it is proved by calculating the rms error of the corner points which is less than 3% which is the standard distance residual variance of LRF.

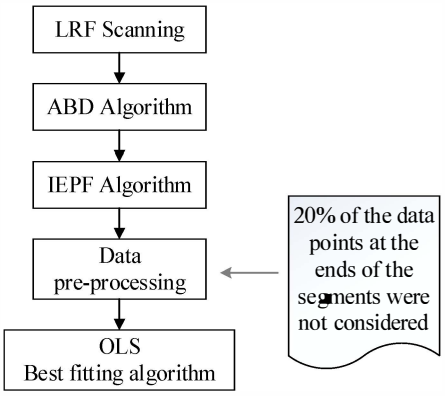


Fig. 7. Flowchart of the segmentation algorithm

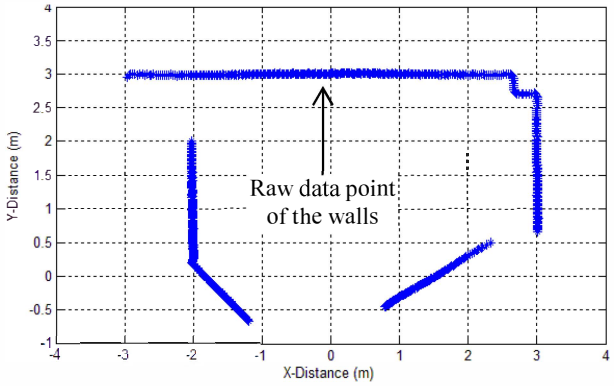


Fig. 8. Single scanning data of the environment

As in the Fig. 9 some of the end points were not detected as the landmarks, but the corners which are created by intersecting two walls have been detected. All the end points of the segments were not considered in the environment because there is no guarantee that they will form corners all the time.

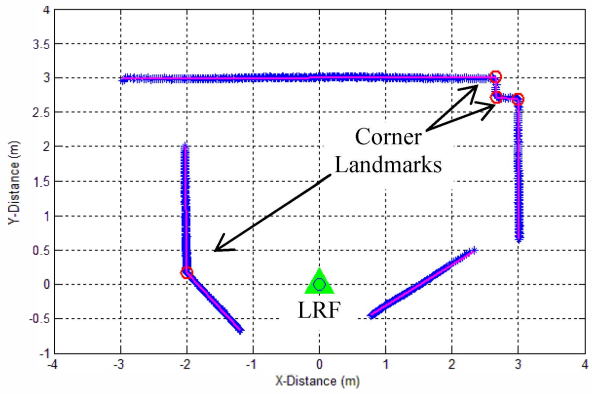


Fig. 9. Results of the segmentation and feature extraction algorithm

TABLE I. ESTIMATED FEATURE POINTS FOR A SINGLE SCAN

Land mark	Coordinates of the corner landmarks		RMS Error %
	True data	*Estimated data*	
C3	(3.0, 2.71)	(2.9879, 2.6894)	1.19
C4	(2.66, 2.71)	(2.6654, 2.7166)	0.43
C5	(2.66, 3.0)	(2.6544, 3.0150)	0.80
C7	(-2.0, 0.20)	(-1.9989, 0.1685)	1.58

Furthermore, experiment was carried out in a different indoor environment (in the corrido of the department) and environmental data were captured at three different points. These extracted data were processed through the segmentation and feature extraction algorithm and line segments and corner landmark points were estimated. Fig. 10 explains the frames of the global map for each scanning data at three different points.

Table II shows the list of detected and estimated corner landmark points for three locations. Some of the landmarks were not detected in each location, but some of them have been detected from two different locations and estimated coordinates are very closer. In this process, repeatedly detected landmarks' coordinates are not same but similar. After the line segments extending and merging process, corner landmarks were estimated again.

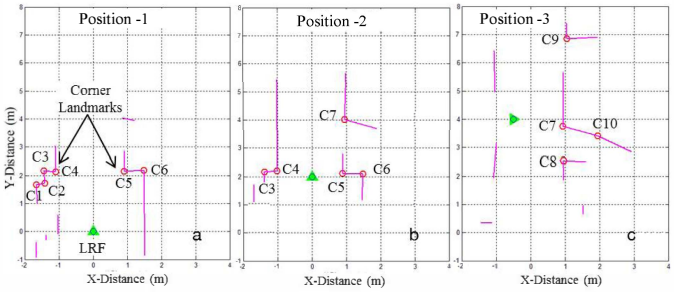
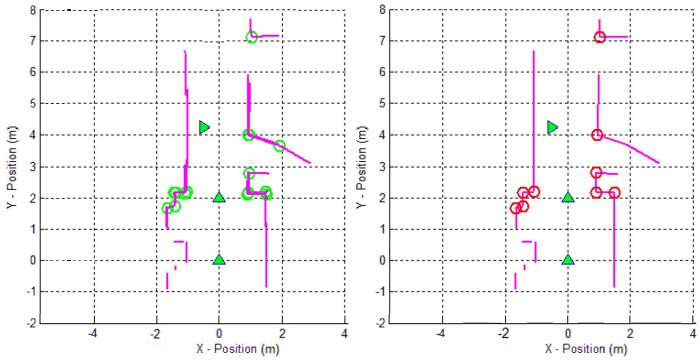


Fig. 10. Frames of the global map and detected corner landmarks

TABLE II. CORNER LANDMARKS DETECTED BY THE ALGORITHM

Land mark	Coordinates of the corner landmarks			
	Position-1 (0,0,0)	*Position- 2 (0,2,0)*	*Position- 3 (-0.5, 4, - π/2)*	*Estimated landmarks*
C1	(-1.658, 1.672)	--	--	(-1.659, 1.672)
C2	(-1.417, 1.728)	--	--	(-1.417, 1.728)
C3	(-1.435, 2.163)	(-1.386, 2.162)	--	(-1.435, 2.163)
C4	(-1.089, 2.141)	(-1.026, 2.205)	--	(-1.089, 2.204)
C5	(0.909, 2.155)	(0.8973, 2.115)	--	(0.909, 2.156)
C6	(1.483, 2.182)	(1.493, 2.103)	--	(1.483, 2.181)
C7	--	(0.947, 4.011)	(0.941, 3.751)	(0.947, 4.004)
C8	--	--	(0.957, 2.545)	(0.918, 2.796)
C9	--	--	(1.047, 6.858)	(1.027 , 7.108)
C10	--	--	(1.949, 3.409)	--

Results of the line segment extending and merging process is shown in Fig. 11 (b) and the line segment map before merging as in (a). It demonstrates that the map after the segment merging process is better than the map generated by using the segment only. The segmentation and merging algorithm has reduced the number of segments which are used to represent the whole environment from 15 to 9 for this three scanning.



(a) (b)

Fig. 11. Generated map of the environment: (a) Before merging (b) After merging

A. *Results of Line Segmentation Approaches*

Experimental analysis of data segmentation method was carried out for different threshold definitions which are explained in section II. Simulation study was first carried out and line segmentation and feature extraction rate (SR) is calculated to evaluate the results. For this evaluation process, in a known environment expected segment length L_i^{exp} and detected length L_i^{det} was used to calculate SR.

$$SR = \frac{\sum_i^k L_i^{det}}{\sum_i^k L_i^{exp}} \cdot 100\% \qquad (11)$$

Different ABD techniques were considered to check the performance of the segmentation algorithm. SR value for the ABD algorithm explained in (6) with slight modification showed a good performance than the other existing methods. The proposed ABD technique for this experiment is given in (12). Compared to the proposed one with the existing methods, it has shown the best segmentation rate. The SR value of the proposed method was 92.4% while the method given in (6) has shown 82.7% and other method has less value than this.

$$D_{thd} = min\{r_n, r_{n+1}\}(1 - 3\sigma_r) \cdot \frac{sin(\Delta\alpha)}{sin(\lambda - \Delta\alpha)} + 3\sigma_r . max\{r_n, r_{n+1}\} \qquad (12)$$

IV. Conclusion and Future Work

A line segmentation and corner landmark detection algorithm is presented in this paper and its performance was evaluated by the experiments. By comparing the evaluation factors for each case and the resulting line segment map the proposed algorithm works well in line segmentation and feature extraction in indoor environments. For continuous detection of the landmarks, algorithm has estimated their coordinates accurately. This algorithm uses stationary LRF data captured at the different known locations for primary implementation and testing assuming that it is similar to the situation when the robot gives localization information correctly. The segment merging process has shown much better performance than the existing methods in the point of the segmentation rate. Finally, we can conclude that the proposed algorithm shows the expected performance and it can be used for mobile robot localization and navigation while extracting the feature online and to enhance the localization.

References

[1] Premebida, Cristiano, and Urbano Nunes. "Segmentation and geometric primitives extraction from 2d laser range data for mobile robot applications." *Robotica* 2005 , pp. 17-25, 2005.

[2] K. Dietmayer, J. Sparbert, and D. Streller, "Model based object classification and tracking in traffic scences from range images", in *International Conference on Information Visualisation,* 2001.

[3] G. Borges and M. Aldon, "Line extraction in 2d range images for mobile robotics," *Journal of Intelligent and Robotic Systems,* vol. 40, no. 3, pp. 267-297, 2004.

[4] Yin, Jingchun, et al. "Graph-based robust localization and mapping for autonomous mobile robotic navigation." *Mechatronics and Automation (ICMA), 2014 IEEE International Conference on. IEEE*, 2014.

[5] He, Xiang, and Zhihao Cai. "Feature extraction from 2D laser range data for indoor navigation of aerial robot." *Chinese Automation Congress (CAC)*. 2013.

[6] Zhou, Jifu, Luis E. Navarro-Serment, and Martial Hebert. "Detection of parking spots using 2D range data." *Intelligent Transportation Systems (ITSC), 2012 15th International IEEE Conference on. IEEE*, 2012.

[7] L. Locchi, and D. Nardi, "Hough localization for mobile robots in polygonal environments," *Robotics and Autonomous Systems*, vol. 40, no. 1, pp. 43-58, 2002.

[8] Certad, Novel, et al. "Study and Improvements in Landmarks Extraction in 2D Range Images Based on an Adaptive Curvature Estimation." *Andean Region International Conference (ANDESCON), 2012 VI. IEEE,* 2012.

[9] Dietmayer, Klaus CJ, Jan Sparbert, and Daniel Streller. "Model based object classification and object tracking in traffic scenes from range images." *Proceedings of IV IEEE Intelligent Vehicles Symposium.* 2001.

[10] Santos, Sérgio, et al. "Tracking of multi-obstacles with laser range data for autonomous vehicles." Proc. 3rd National Festival of Robotics Scientific Meeting (ROBOTICA). 2003.

[11] Lee, Kenneth Jay. "Reactive navigation for an outdoor autonomous vehicle." *Master's Thesis, University of Sydney, Sydney, Australia* (2001).

[12] M. Adams, "On-line gradient based surface discontinuity detection for outdoor scanning range sensors," in *Proc. Of IEEE/RSJ International Coference on Intelligent Robots and Systems,* vol. 3, pp. 1726-1731, 2001.

[13] M. Liu, X. Lei, S. Zhang, B. Mu, "Natural Landmark Extraction in 2D Laser Data based on Local Curvature Scale for Mobile Robot Navigation," *in Proc. Of the IEEE International Conference on Robotics and Biomimetics*, pp. 525-530, 2010.