An Implementation of Typical-Obstacle Detection and Recognition with Laser Range Finder

Quan Qiu\textsuperscript{1,2} 
1. State Key Laboratory of Robotics 
Shenyang Institute of Automation, Chinese Academy of Sciences 
Shenyang, 110016 China 
quuquan@sia.cn

Jianda Han\textsuperscript{1} 
2. Graduate University of Chinese Academy of Sciences 
Beijing, 100049 China 
jdhan@sia.cn

Abstract—This paper presents an implementation of detecting and recognizing typical obstacles with laser range finders. Mounting the laser range finder with a pitch angle on top of the robot, we can obtain the height information of the obstacle. Techniques, such as clustering, line segment extraction, coordinate transformation, line segment matching, are then applied on the scan data analyzing to detect and recognize the platforms and ditches. Experiments are conducted to validate our implementation.

Keywords—typical-obstacle; detection; recognition; laser range finder

I. INTRODUCTION

With an enhanced obstacle-negotiation capability [1], robots nowadays lay higher requirements on the environmental perception step, hoping that it can not only report the existence of the obstacles but also recognize obstacles with different travel cost levels, requiring reconfiguration or not for example. To reach the effective perception level, different kinds of sensors are employed to collect environmental information, such as sonar, laser scanner, and video camera. Among them, laser range finder (LRF) is a popular choice for many researchers, because of its perfect performances in real-time applications and its high measurement accuracy.

Now, LRF has been widely used in detecting and classifying obstacles, including the applications on navigation [2], SLAM [3], traffic warning system [4] and others. Also, LRF can cooperate with video cameras to fulfill the detection and classification [5]. According to the existing algorithms, an LRF scan will be pretreated by the clustering and feature extraction techniques to obtain the rough description of the objects [6], then a variety of approaches can be used to recognize or classify the objects: voting schemes [7], a method based on “heuristic” rules [8], and multi-hypotheses [9], etc. The first two methods have the disadvantage of not having a self-consistent mathematical framework in order to support its stability and consistency. The multi-hypotheses method presented is often based on features tracking by means of a Kalman Filter, whose main drawback is a high computational cost and the inconvenient of managing many hypotheses.

This work was supported in part by the National Natural Science Foundation of China under Grant 60775056 and 60705028, and supported by the National High Technology Research and Development Program of China under Grant 2006AA04Z215 and 2007AA04Z250.
\( \mathbf{r}_g = (x_r, y_r, 0, 1)^T \) \hspace{1cm} (1)

We can represent \( \mathbf{r}_g \) in the global coordinate system \( G \) by multiplying a transform matrix \( T \)

\[
T = \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & \cos(\phi) & \sin(\phi) & 0 \\
0 & -\sin(\phi) & \cos(\phi) & Z \\
0 & 0 & 0 & 1
\end{bmatrix}
\] \hspace{1cm} (2)

\[
\mathbf{r}_g = T \cdot \mathbf{r}_s = (x, y, z, 1)^T
\] \hspace{1cm} (3)

in which \( \phi \) is the pitch angle, \( Z \) is the vertical distance from the laser source to the bottom surface of the robot, \( z \) is the “Height” of the detected obstacle. Readers can find the physical meanings of the parameters in Figure 2.

III. IMPLEMENTATION

The whole implementation process of detecting the obstacles and recognizing the platforms or ditches can be divided into five steps: 1) clustering the scan points, 2) extracting line segments from the clusters, 3) obtaining line segments in the 3D space, 4) matching the corresponding lines in the scans of adjacent time stamps, and 5) recognizing the obstacles and delivering the dimensional descriptions.

A. Clustering

Clustering can find the suspect obstacles from the rough scan data. As the scan data is a series of distances queued in an angular ascending order, the relative position of the adjacent two points in the scan can be used to partition different clusters. The common method is that if the distance between two adjacent points exceeds a preset threshold, we can infer that the two scan points are dispatched on different objects and can be divided into different clusters. We denote the positions of the two adjacent points in the scanner coordinate system by \((x_{i}, y_{i})\) and \((x_{i+1}, y_{i+1})\) respectively. Then the clustering rule can be given by

\[
\sqrt{(x_{i+1} - x_{i})^2 + (y_{i+1} - y_{i})^2} \leq D_{thr}
\] \hspace{1cm} (4)

where \(D_{thr}\) is the preset threshold and is determined according to the working environment of the scanner. Figure 3 gives an clustering example. Figure 3(a) is the rough scan map, and figure 3(b) is the clustering result in which all the adjacent clusters are printed in different colors.

Figure 2. Obtaining the height information

The height information of the object is an important factor for recognizing platforms and ditches in our implementation.

B. Line Segment Extraction from the Cluster

Clustering can categorise points belonging to different objects, but fails in the intensive description of the inner structures of the point set. Then we need to do line segment extraction within a cluster to achieve the intensive description. Line segment extraction in cluster can be divided into two parts: cluster splitting and line segment fitting.

The points of one cluster may be fitted to different line segments. Cluster splitting is the process that splits a cluster into small sub-clusters whose inner points can only be fitted to one line segment. A line segment can be represented by \(l(\rho, \theta)\) from following line equation
The splitting is started with a line segment growing process. First, the first two points of the cluster will be used to compute a line presentation \( l(\rho, \theta) \). Second, the segment growing process starts from the third point. A judging rule presented in [6] will be employed to determine whether or not a new can be added into the line segment. If the rule is not met, the growing is stopped and the points added into the line segment will form a new sub-cluster. These steps will be repeated until the last point of the cluster is reached. And finally a cluster will be split into one or more sub-clusters.

The points of one sub-cluster will be fitted to a line segment with a linear regression method [6]. If we denote a scan point in the scanner coordinate system by \( p_{X_i}(x_i, y_i) \), a cluster can be expressed as a point set \( \{X_i(x_i, y_i)\}_{i=1}^{n} \). The regression method can be implemented by four steps.

1) Compute the regression parameter set \((R_x, R_y, R_{xx}, R_{yy}, R_{xy})\)

\[
R_x = \sum_{i=1}^{n} x_i; \quad R_y = \sum_{i=1}^{n} y_i; \\
R_{xx} = \sum_{i=1}^{n} x_i^2; \quad R_{yy} = \sum_{i=1}^{n} y_i^2; \quad R_{xy} = \sum_{i=1}^{n} x_i y_i.
\] (6)

2) Compute \( N \), which represents the width of the cluster along the x axis

\[
N = R_{xx} n^2.
\] (7)

3) Compute \( m \) and \( q \), which can be used to form another representation of a line as \( y = mx + q \)

\[
m = \frac{T}{N}, \quad q = \frac{(R_y - mR_x)}{n} \quad \text{for} \quad T = nR_{xx}R_y - R_{xy}.
\] (8)

4) Compute \( \rho \) and \( \theta \)

\[
\rho = \frac{q}{\sqrt{m^2 + 1}}, \quad \theta = \arctg \left( \frac{q}{m} \right)
\] (9)

Figure 5 shows the process of cluster splitting and line segment extraction.
C. Obtaining Line Segments in the 3D Space

To fulfill the obstacle recognition task, we need to transform the line segments from the 2D scanner coordinate system into the 3D global coordinate system, and use the height information to detect and recognize the obstacles. As we only consider the typical structured obstacles, such as platforms and ditchers, we assume all the line segments in the space are parallel to the horizontal plane. Then the transformation of a line segment can be simplified to the transformation of the begin point and the end point. After the transformation of the begin and end point is done, the parameter set of the new line segment in the space can be defined as \{\rho, \theta, h, len\}. The meanings of the parameters are shown in Figure 6. Figure 7 shows a scan map after line segment transformation, which comes from the line extraction result of Figure 5(c).

D. Matching the Corresponding Lines in the Scans of Adjacent Time Stamps

The information contained in one single usually is not enough for obstacle recognition, so we need to analyze the changing line segment of the corresponding obstacle in the adjacent two or more time stamps. Then matching the line segments in different scans becomes very important. In [9], a Kalman Filter is employed to do the matching. Here, we just use the parameters of the space line segment to find the corresponding line segments. For two line segments \{\rho_1, \theta_1, h_1, len_1\} and \{\rho_2, \theta_2, h_2, len_2\} in two scans respectively, if they meet all the rules in (10), they are considered as corresponding line segments.

\[
\begin{align*}
|\rho_1 - \rho_2| < \rho_{\text{thres}}, \\
|\theta_1 - \theta_2| < \theta_{\text{thres}}, \\
|h_1 - h_2| < h_{\text{thres}}, \\
|len_1 - len_2| < len_{\text{thres}}
\end{align*}
\]  

(10)

where \rho_{\text{thres}}, \theta_{\text{thres}}, h_{\text{thres}}, and len_{\text{thres}} are the predefined thresholds. A matching example is shown in Figure 8.

E. Recognizing the Obstacles and Delivering the Dimensional Descriptions

We choose platforms and ditchers as our typical obstacles. A platform is a flat with a higher height value than the ground, while a ditch is a flat with a lower height value than the ground. Figure 9 show a platform and a ditch.
Here, we proposed a strategy named by “invariant height” strategy to recognize the platform and ditch. This strategy comes from the flat feature of the platforms and the ditches. Plus or minus, if the obstacle’s height value lies in a small and invariable region through a series of continuous time stamps, we can infer that a platform or a ditch is detected and recognized.

IV. EXPERIMENTAL RESULTS

Experiments were conducted to test the feasibility of our implementation. A 5m×1.5m×0.4m cuboid took the role of both the platform and the ditch. When the robot moved on the ground, the cuboid could form a platform together with the ground surface. When the robot moved on the cuboid, it also could form a ditch together with the ground surface. In the experiment, we put the platform or the ditch in front of the robot, and let the robot move towards with a working laser scanner. The scan frequency of the scanner is about 5Hz. Then the typical obstacle could be recognized in the marching process. We repeated the platform detection and ditch detection five times, respectively. The obtained dimensional descriptions of the obstacles are listed in Table I and Table II.

TABLE I. DIMENSIONAL DESCRIPTIONS OF THE PLATFORM

<table>
<thead>
<tr>
<th>Experiment No.</th>
<th>Distance(m)</th>
<th>Height(m)</th>
<th>Width(m)</th>
<th>Orientation (rad)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2.641</td>
<td>0.391</td>
<td>1.491</td>
<td>1.571</td>
</tr>
<tr>
<td>2</td>
<td>2.623</td>
<td>0.395</td>
<td>1.487</td>
<td>1.574</td>
</tr>
<tr>
<td>3</td>
<td>2.635</td>
<td>0.389</td>
<td>1.489</td>
<td>1.578</td>
</tr>
<tr>
<td>4</td>
<td>2.638</td>
<td>0.392</td>
<td>1.493</td>
<td>1.590</td>
</tr>
<tr>
<td>5</td>
<td>2.632</td>
<td>0.394</td>
<td>1.492</td>
<td>1.582</td>
</tr>
</tbody>
</table>

TABLE II. DIMENSIONAL DESCRIPTIONS OF THE DITCH

<table>
<thead>
<tr>
<th>Experiment No.</th>
<th>Distance(m)</th>
<th>Height(m)</th>
<th>Width(m)</th>
<th>Orientation (rad)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2.641</td>
<td>0.392</td>
<td>1.485</td>
<td>1.579</td>
</tr>
<tr>
<td>2</td>
<td>2.623</td>
<td>0.390</td>
<td>1.489</td>
<td>1.581</td>
</tr>
<tr>
<td>3</td>
<td>2.635</td>
<td>0.388</td>
<td>1.493</td>
<td>1.592</td>
</tr>
<tr>
<td>4</td>
<td>2.638</td>
<td>0.397</td>
<td>1.491</td>
<td>1.585</td>
</tr>
<tr>
<td>5</td>
<td>2.632</td>
<td>0.391</td>
<td>1.482</td>
<td>1.593</td>
</tr>
</tbody>
</table>

From Table I and Table II, we can infer that our implementation can detect and recognize the platform and the ditch reliably, and the errors of the presented dimensional description are less than 2cm.

V. CONCLUSION

An implementation of typical obstacle detection and recognition on a robot equipped with laser range finder is proposed in this paper. The height information of the obstacles is obtained by mounting the LRF with a downward pitch angle and transforming the data from the 2D scanner’s coordinate system into the 3D global coordinate system. Clustering is first used to categorise points of different objects. Line segment extraction is then applied to give inner descriptions of each cluster. After transforming the line segments from the scanner coordinate into the global coordinate, a line-segment matching strategy is executed to find the corresponding line segments in different scan maps. Finally, the “invariant height” strategy is employed to recognize the platforms and ditches which are the main typical obstacles in the implementation. Experimental results show that the implementation delivers a satisfying real-time performance and works reliably with a high accuracy for the dimensional description of the obstacle.

ACKNOWLEDGMENT

The authors would like to thank Chunguang Bu, Jingli Lu and Fei Chang for the heuristic discussion, implement assistance, and kind encouragement.

REFERENCES