Ground Segmentation Algorithm Based on Point Cloud Data

Honghui Mu
Changchun Automobile Industry Institute, Changchun 130013, China,
muhonghui@foxmail.com,
corresponding author

Yanli Yang
Changchun Automobile Industry Institute, Changchun 130013, China
yangyanli1010w@163.com

Longkai Liang
Jilin University, Changchun 130012, Jilin, China
jeasonllk@qq.com

ABSTRACT

Aiming at the problem of under segmentation between ground point cloud and multi-target object point cloud, a ground segmentation algorithm for point cloud data based on multi-region segmentation is proposed in this paper. In the process of LIDAR point cloud filtering, the method of mapping 3D point cloud to grid map is used to detect and filter the ground point cloud. The algorithm uses plane fitting to detect the ground point cloud in each grid, and filters the ground point cloud in the grid to retain the non-terrestrial point clouds. The algorithm can reduce the possibility of under segmentation of ground point cloud effectively, reduce the amount of calculation of subsequent target detection, and improve the accuracy.

CCS CONCEPTS

• **Computing methodologies**; • **Modeling and simulation**; • **Model development and analysis**;

KEYWORDS

point cloud data, LIDAR, segmentation, styling

ACM Reference Format:
Honghui Mu, Yanli Yang, and Longkai Liang. 2021. Ground Segmentation Algorithm Based on Point Cloud Data. In *2021 3rd International Conference on Artificial Intelligence and Advanced Manufacture (AIAM2021), October 23–25, 2021, Manchester, United Kingdom.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3495018.3495281

1 INTRODUCTION

In robot perception, the surface that the robot can cross is represented as the generalized ground [1], unmanned vehicle is a kind of special wheeled mobile robot, which can cross the "ground" more widely than the narrow sense robot system, it's includes regular ground and irregular ground in urban street, suburban road, rural road and other application scenarios [2]. Processing a large number of point cloud data requires huge computing power. In addition, due to the discontinuity of point cloud, it is necessary to combine geometric points into a semantic meaningful point cloud cluster. Therefore, it is necessary to preprocess the original point cloud collected by LIDAR to eliminate unnecessary elements and reduce the data dimension of the target before entering the detection process [3]. The ground removal process is mainly to distinguish the untraceable objects such as the ground and curbs from the unique objects of interest such as cars, bicycles and pedestrians.

The existing ground segmentation algorithms can be roughly divided into grid based and object-based methods, [4] introduced the idea of offline training and online detection to detect ground point cloud, which improves the detection accuracy and online detection efficiency of ground point cloud. A significant occlusion approach was proposed in literature [5], which combined channel-based ground classification with network-based ground height representation to deal with false spatial measurements, and used inter-channel correlation to compensate for lost data.

In this paper, aiming at the under segmentation phenomenon in the traditional random sampling consensus algorithm (RANSAC) for ground detection, piecewise surface estimation algorithm based on multi region strategy and velodyne LIDAR scanning behavior is proposed, the algorithm combines RANSAC algorithm and least square method (LSM), which can remove the invalid point cloud segmentation effectively.

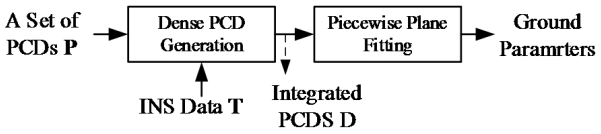

Figure 1: Piecewise ground surface estimation

2 SEGMENTED SURFACE ESTIMATION

The point cloud has the characteristics of space, disorder, sparsity and rotation invariance, the dense model of the scene is formed by combining the time series of 3D LIDAR data and GPS Aided INS positioning data [6]. The piecewise surface estimation algorithm based on "multi region" strategy and velodyne LIDAR scanning behavior is adopted [7], RANSAC is used to fit the limited number of planes of the road and its adjacent area. The segmented surface estimation model is shown in figure 1

2.1 Ground Estimation

The generation process of dense point clouds is shown in table 1, p_i represent the 3D point cloud data in the current time i [8], the first m point cloud data can be expressed as $P=\{P_{i-m},\ldots,P_i\}$, $T=\{T_{i-m},\ldots,T_i\}$ is defined the set of vehicle attitude parameters and define function $EI(R_K,T_K)= R_K \times R_{K+}\ t_K$ to represent the transformation from self-vehicle to world coordinate system.

AIAM2021, October 23–25, 2021, Manchester, United Kingdom

ACM ISBN 978-1-4503-8504-6/21/10…$15.00
https://doi.org/10.1145/3495018.3495281

Table 1: dense point cloud generation

Algorithm 1: Dense point cloud generation.
1: Input: PCDs: **P** and Ego-vehicle Poses: **T**
2: Output: Dense PCD:
3: for scan k=i-m to i **do**
4: T_{k1}←ICP(BGF(GI(P_i,T_i),BGF(GI(P_k,T_k)))
5: D←Merge(P_k, T_k, T_{k1})
6:end for

The BGF (Box Grid Filter) is used to down sample PCDs, divide the space into voxels, q is the average value of each voxel, according to certain constraints, ICP (Iterative Closest Point Algorithm) is used to find the nearest point (p_i, q_i) in the target point cloud p and source point cloud Q to be matched [9]. The optimal matching parameters R and T are calculated to minimize the error function. The error function $E(R, t)is$:

$$E(R,t) = \frac{1}{n}\sum_{i=1}^{n}\left\|q_i - (Rp_i + t)^2\right\| \tag{1}$$

Where n is the number of nearest neighbor pairs, p_i is a point in the target point cloud P and q_i is the nearest point corresponding to q_i in the source point cloud Q, R is rotation matrix, t is translation vector.

Merge function is used to convert the PCDS to the current coordinate system of the car by using (2), and the dense point cloud generation algorithm is shown in table 1

$$N_i = \mathrm{U}_{k=i-m}^{i} R_i^{-1} \times ((\hat{R} \times P_k + \hat{t}_k) - t_i) \tag{2}$$

Dense PCD generated by ICP algorithm is shown in figure 2, the red rectangle in the image above represents a traffic post, the results obtained by using BGF and ICP steps to further align continuous PCDs and reduce positioning errors is shown in lower right corner.

2.2 Piecewise Plane Fitting

The spatial 3D point cloud data is mapped into the grid map, and the ground point cloud is fitted by the point cloud data in the grid, so as to improve the stability of the ground extraction. Firstly, the excitation point cloud is segmented into multiple regions [10], and then RANSAC is used to fit the ground point cloud for each small region. At the same time, the least square is used to speed up the fitting speed.

1) Gridding of point cloud: The point cloud data is regarded as a whole block, surrounded by the smallest cuboid, and the minimum side length of the cuboid is set according to the actual size of the point cloud. The cuboid is divided into a three-dimensional grid to form a multi-region segmentation of point cloud data [11].

- Calculate the maximum and minimum values of the three coordinate axes of the point set P={P_1,P_2,. . .,P_i}:

$$\begin{aligned} X_{max} &= \mathrm{MAX}(x_1, \cdot s, x_N), X_{\min} = \mathrm{MAX}(x_1, \cdot s, x_N) \\ Y_{max} &= \mathrm{MAX}(x_1, \cdot s, x_N), Y_{\min} = \mathrm{MAX}(x_1, \cdot s, x_N) \\ Z_{max} &= \mathrm{MAX}(x_1, \cdot s, x_N), Z_{\min} = \mathrm{MAX}(x_1, \cdot s, x_N) \end{aligned} \tag{3}$$

- Determine gridded side length, the gridded side length R determines the number of points in each grid and the computational efficiency. The number of grids is inversely proportional to the rasterized side length R, the dimension calculation formula of point cloud grid is shown as follows:

$$\begin{aligned} D_x &= [(X_{\max} - X_{\min})/R] \\ D_y &= [(Y_{\max} - Y_{\min})/R] \\ D_z &= [(Z_{\max} - Z_{\min})/R] \end{aligned} \tag{4}$$

- Calculate the index of each point after rasterization. The point cloud after rasterization is coded to determine the number of the grid where each point is located. The index h of each point in the grid is calculated as follows:

$$\begin{aligned} h_x &= [(x - X_{\min})/R] \\ h_y &= [(y - Y_{\min})/R] \\ h_z &= [(z - Z_{\min})/R] \\ h &= h_x + h_y * D_x + h_z * D_x * D_y \end{aligned} \tag{5}$$

2) Multi-region segmentation: Focusing on the vehicle, the geometry of the Velodyne LIDAR scan is followed to generate a finite set of areas on the ground as shown in figure 3. Set the section of the nearest area to the host vehicle as S_0, let the scanning radius of LIDAR λ_0=5, and the size of the remaining area increases continuously. According to the model shown in figure 3, the edge radius of slice S_K is given by the following tangent function:

$$\lambda_k = h \cdot tan(\alpha_0 + k \cdot \eta \cdot \Delta\alpha), \qquad \{k : 1, \cdot s, N\} \tag{6}$$

Figure 2: Possess Rich Marine Resources Dense PCD generated by ICP algorithm

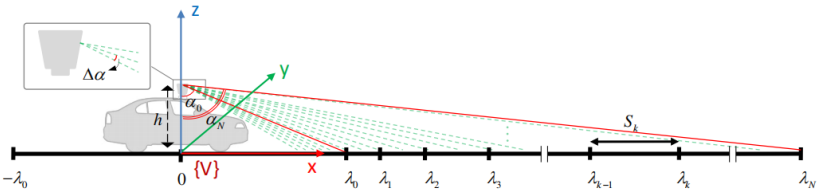

Figure 3: Diagram of variable-size ground slices η =2

Where α_0=arctan(λ_0/h), h is the elevation from LIDAR to ground, N= (α_N-α_0)/η•$\Delta\alpha$.is the total number of slices, $\Delta\alpha$is the angle increment between scan lines, $\eta$is a constant, is the rounded down function.

3) RANSAC plane fitting: RANSAC is an iterative algorithm to fit the parameters of mathematical model, the advantage of the algorithm is that it has good robustness to the data with outliers [12-10]. The RANSAC algorithm is applied to the extraction of ground point cloud, which belongs to outlier point cloud data except ground point cloud. The specific implementation steps are as follows:

Three points are randomly selected in the point cloud, and the normal vector of the plane is calculated as follows:

$$\vec{n} = (P_2 - P_1) \times (P_3 - P_1) \tag{7}$$

Where P_i=(x_i, y_i, z_i).

Calculate the distance from any point in the point cloud to the plane:

$$d_i = \frac{\vec{n}^T (P_i - P_1)}{\|\vec{n}\|_2} \tag{8}$$

Where p_i is any point in the point cloud, i=1,2,3. . .N.

Set the threshold value to extract the normal point cloud data, save the qualified point cloud, and record the number of points.

Repeat steps 1) ~ 3) T times, then save the point cloud with the largest number of points, and use the least square method to fine tune the ground point cloud to fit the accurate plane parameters.

Maximum number of iterations as in:

$$T = \frac{\log(1-p)}{\log(1-(1-e)^s)} \tag{9}$$

Where e is Ratio of outliers in point cloud data, p is the probability of selecting normal point at least once, S is the number of selected points per iteration.

Least square method: Due to the randomness of RANSAC fitting, and when the ratio e of outlier point cloud is set incorrectly, inaccurate ground point cloud is extracted even within the maximum number of iterations. Set an expected normal ratio E=1-e, when the ratio of the total number of ground point clouds extracted by RANSAC is greater than E, the iteration is terminated. Finally, according to the fitted plane model parameters, the ground point cloud is extracted from each grid point cloud. The ground model parameters are shown in (10):

$$\vec{x} = (A^T A)^{-1} A^T \vec{b} \tag{10}$$

Where x is the plane normal vector $\boldsymbol{x}$=[a,b,c]^T, A is the matrix composed of point clouds extracted by RANSAC, A={P_1,. . .,P_S}^{T}, and S<N,$\boldsymbol{b}$=[-1,-1,-1]^T is constant coefficient of ground model. The segmented surface estimation algorithm is shown in table 2

Table 2: piecewise ground surface estimation

Algorithm 2 :Piecewise Ground Surface Estimation.

1: Input: Dense PCD D
2:Output: Ground Model G={G_1,,G_N}
3: for slice k=1 to N **do**
4: S_k←*Slice*(D)
5: m_k←Gate(S_k)
6: D_k←RANSA(m_k)
7: G_k←LSM(D_k)
8:end for

3 GROUND OBSTACLE SEGMENTATION

Obstacle segmentation model is shown in figure 4, the original PCD is represented by filled gray circles, set the fitting plane model as G_k, and the orange dotted line indicates the threshold lower limit L_{min} and the upper threshold L_{max}, where L_{min}=$L_{25\%}$-0.5IQR, L_{max}=$L_{75\%}$, $L_{25\%}$is median of the lower half of the data, $L_{75\%}$ is median of the upper half of the data, IQR= $L_{75\%}$ - $L_{25\%}$ is intermediate quintile [15]. The range of data points in the black dotted rectangle is $L_{min<Z<}$ L_{max}, the green solid line is the plane estimated by RANSAC, the green dotted line indicates the continuation of the fitting plane of S_k in the slice S_{k+1}.δ_{Zk+1}=| Z_{k+1}- Z_k| is the distance between two continuous planes S_k and S_{k+1}, Z_k is the value of G_k at the edge of the slice, the z value for G_k can be computed by representing the plane equation as: z=-(a_k/c_k)x-(b_k/C_k)y-(d_k/C_k), δ_{Zk+} is the angle between two continuous planes, its value as (11).

$$\delta\psi_k = \arctan\left|\frac{n_{k-1} \times n_k}{n_{k-1} \cdot n_k}\right| \tag{11}$$

n_k and n_{k-1} are the unit normal vectors of plane G_k and G_{k-1}. Dashed magenta lines show the threshold be used for the ground obstacle segmentation task. Points under dashed magenta lines are considered as ground points.

Let P_0 be any point in the given surface plane, calculate the distance from the point p to the plane G_k by dot product d=(p-p_0)•n_k, the segmentation threshold is set as d_{min}, the part of the ground plane lower than point d_{min} is removed, and the remaining points represent the points of obstacles.

In this paper, the threshold d_{min} is determined by experimental visualization, standard deviation of the distance from all points to the plane can be defined σ:

$$\sigma = \sqrt{\frac{\sum_{i=1}^{N}(d_i - \bar{d})^2}{N-1}}$$
$$\bar{d} = \frac{1}{N}\sum_1^N d_i \tag{12}$$

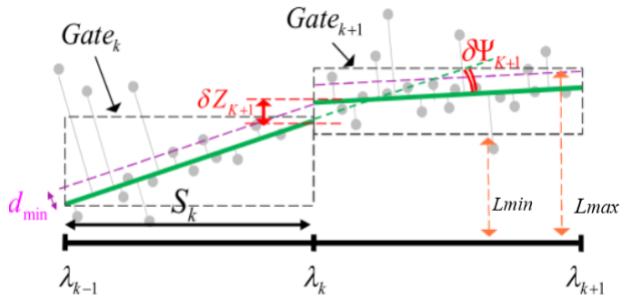

Figure 4: The ground obstacle segmentation model

Table 3: Main parameters used in the ground estimation

m	η	h	$\Delta\alpha$
6	6	1.73	0.4°

When $d_i > d_{\min} = 2\sigma$, this point is considered as an outlier and is not extracted as a ground point cloud. On the contrary, the effective value points are extracted as ground point clouds.

4 EXPERIMENT AND RESULT ANALYSIS

4.1 Experiment Index

The operating system of the experimental environment is 64 bit Ubuntu 16.04.3 LTS, and the processor is inter (R) core ™ i7-6700 CPU, 32G memory, GTX-1080 graphics card.

KITTI data set, the largest outdoor real scene driverless data set in the world, is selected as the experimental data set, which contains real data of urban, rural, highway and other different scenes. This experiment selects a KITTI point cloud data to compare the number of ground point clouds extracted by the algorithm and the visualization, in which the grid size is 25 meters.

4.2 Experimental Results and Analysis

The main parameters used in the ground estimation are shown in table 3

The ground estimation error is calculated by the average distance from the base point of GT 3D BBS to the estimated ground. Specifically, the mean of displacement error (MDE) in the i−th frame is defined as:

$$mDE^{(i)} = \frac{1}{M}\sum_{k=1}^{M}\left|(P_k{}^G - P)\cdot n\right| \tag{13}$$

Where $P^G{}_k$ denotes the base of the GT 3D-BB for the k−th object, Mis the total number of objects in i−th frame, the variables P and n are the points and unit normal vectors that define the corresponding surfaces respectively. The MDE of the sequence is obtained by (14):

$$mDE = \frac{1}{N}\sum_{i=1}^{N} mDE^{(i)} \tag{14}$$

i is the sum of the total frames of all sequences, MDE is calculated for different integral frame numbers m and η, the results are shown in figure 5. By m=6 and η =6, the minimum MDE is 0.086.

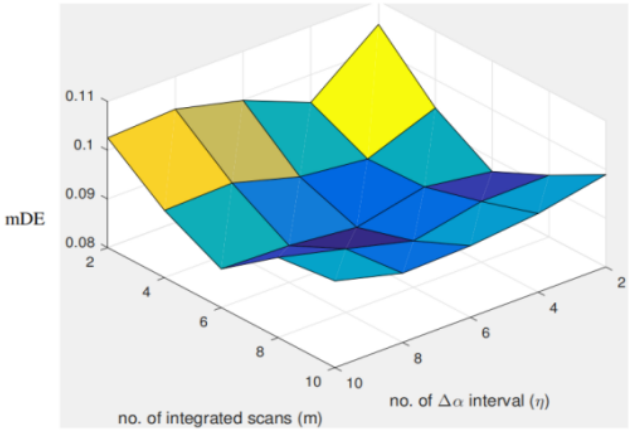

Figure 5: Evaluation of the proposed ground estimation algorithm based on MDE

Table 4: Comparison of two algorithms

Algorithm	Total	Extracted	Time (ms)
RANSAN	12383	59384	264
MRSA	12383	64962	269

The filtering effect of the ground point cloud by using the RANSCAN algorithm and the algorithm in this paper is shown in figure 6

Red in the graph represents the segmented ground point cloud, and RANSCAN algorithm has the phenomenon of under segmentation for the marked position in the graph. Multi region segmentation algorithm (MRSA) divides the ground point cloud into multiple regions for segmentation, which effectively alleviates the under segmentation phenomenon caused by uneven road surface and slope.

It can be seen from table 4 that the number of point clouds extracted based on the RANSAC algorithm is far less than the multi-region segmentation algorithm, because the RANSAC algorithm is random and the phenomenon of under segmentation of ground point clouds is accidental. The algorithm proposed in this paper uses raster processing, so that the ground point cloud is segmented, which can reduce the randomness. As can be seen from figure 6 (b), the algorithm can extract the ground point cloud very stably.

5 CONCLUSION

This paper focuses on the ground detection process in the process of automatic driving and the shortcomings of traditional RANSAC ground detection algorithm, based on the improvement of raster map mapping algorithm, a multi- region based RANSAC ground segmentation algorithm is proposed. The algorithm divides the ground point cloud into multiple regions for segmentation, which alleviates the phenomenon of under segmentation caused by uneven road surface and slope effectively.

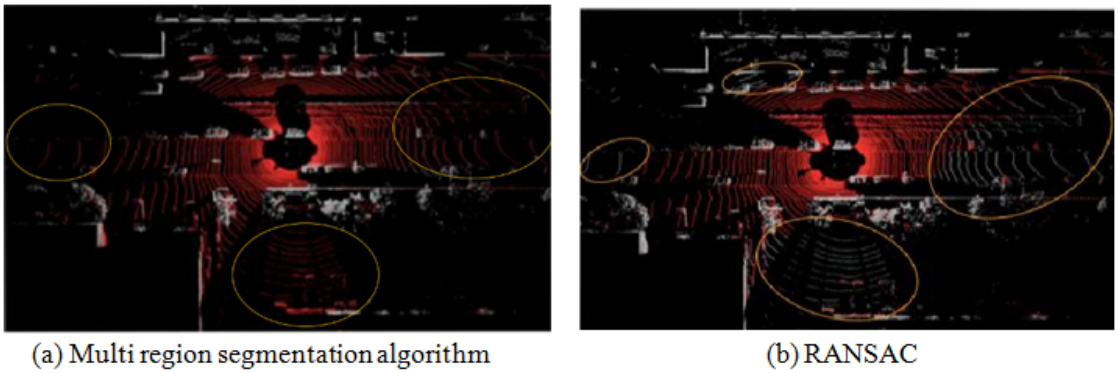

Figure 6: Filtering effect picture of ground point cloud

REFERENCES

[1] Shi S, Wang X, Li H. PointRCNN: 3D Object Proposal Generation and Detection From Point Cloud[C] 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2020,770-779.

[2] Lang A H, Vora S, Caesar H, *et al.* PointPillars: Fast Encoders for Object Detection From Point Clouds[C] 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2019, 12689-12697.

[3] Ohgushi T, Horiguchi K, Yamanaka M. Road Obstacle Detection Method Based on an Autoencoder with Semantic Segmentation[C]// Asian Conference on Computer Vision. Springer, Cham, 2020.

[4] Samples M, James M R. Learning a Real-Time 3D Point Cloud Obstacle Discriminator via Bootstrapping[J]. ISPRS Journal of Photogrammetry and Remote Sensing, 2013,12 (3):14-21.

[5] Sultana M, Mahmood A, Javed S, *et al.* Unsupervised deep context prediction for background estimation and foreground segmentation[J]. Machine Vision and Applications, 2019.

[6] Nicola Bernini, Massimo Bertozzi, Luca Castangia, Marco Patander, and Mario Sabbatelli. Real-time obstacle detection using stereo vision for autonomous ground vehicles: A survey. In Intelligent Transportation Systems (ITSC), 2014IEEE 17th International Conference on, pages 873–878. IEEE, 2014.

[7] Song Q, Cui Z, Liu P. An Efficient Solution for Semantic Segmentation of Three Ground-based Cloud Datasets[J]. Earth and Space Science, 2020, 7(4).

[8] Fan M, Jung S W, Ko S J. Highly Accurate Scale Estimation from Multiple Keyframes Using RANSAC Plane Fitting With a Novel Scoring Method[J]. IEEE Transactions on Vehicular Technology, 2020.

[9] Chen D J, Chien J T, Chen H T, *et al.* Unsupervised Meta-learning of Figure-Ground Segmentation via Imitating Visual Effects[J]. 2018.

[10] Hofmann M, Tiefenbacher P, Rigoll G. Background segmentation with feedback: The Pixel-Based Adaptive Segmenter[C]// Computer Vision & Pattern Recognition Workshops. IEEE, 2012.

[11] Steinberg E, Prilutsky Y, Corcoran P, *et al.* Foreground/background segmentation in digital images with differential exposure calculations[J]. US, 2012.

[12] Geiger A, Lenz P, Stiller C, *et al.* Vision meets robotics: The KITTI dataset[J]. International Journal of Robotics Research, 2013, 32(11):1231-1237.

[13] Lafarge F, Mallet C. Creating Large-Scale City Models from 3D-Point Clouds: A Robust Approach with Hybrid Representation[J]. International Journal of Computer Vision, 2012, 99(1):69-85.

[14] Klokov R, Lempitsky V. Escape from Cells: Deep Kd-Networks for the Recognition of 3D Point Cloud Models[C]// 2017 IEEE International Conference on Computer Vision (ICCV). IEEE, 2017.

[15] Awwad T M, Zhu Q, Du Z, *et al.* An improved segmentation approach for planar surfaces from unstructured 3D point clouds[J]. Photogrammetric Record, 2010, 25(129):5-23.