Obstacle Recognition Based on Multilayer LiDAR

Zhe Xu[1,a] and Xuhui Ran[1,b]

[1]Information Faculty, Beijing University of Technology, Beijing, China
a. 953108340@qq.com, b. ranxuhui33@163.com

Keywords: Driverless vehicle, LiDAR, environmental awareness, Dempster-Shafer Theory, grid map.

Abstract: Aiming at the problem that four-layer LiDAR is difficult to identify neighboring obstacles in the obstacle recognition of complex road environment, an obstacle recognition algorithm considering both adaptive and real-time characteristics is proposed. The algorithm clusters obstacles detected based on Dempster-Shafer Theory (DST) and conflict coefficients in a grid map through segment based connected region labeling algorithm, which reduces the number of neighborhood searches and increases the real-time performance of the algorithm. The Borges distance threshold is improved by combining the characteristics of obstacle data, and adaptive recognition is used to optimize the recognition of neighboring obstacles through the improved distance threshold and the distribution characteristics of obstacle data. The experimental results show that the algorithm can significantly improve the recognition of neighboring obstacles and the real-time performance of the algorithm meets the real-time requirements of detecting obstacles when unmanned vehicles are driving.

1. Introduction

With the continuous development of artificial intelligence technology, self-driving cars have become a hotspot for scholars in various countries as a new direction for the development of automobiles. Among them, environmental awareness has received widespread attention as an important part of driverless car technology. Self-driving cars mainly use sensors such as cameras, LiDAR, and GPS to sense the surrounding environment in real time to meet the requirements of detecting obstacles. At present, researches on obstacle detection at home and abroad mainly focus on computer vision-based methods and LiDAR-based methods. However, the method based on computer vision is easily affected by light, which makes the detection effect poor. Compared with other sensors, LiDAR sensors have the advantages of high measurement accuracy, fast response speed, strong anti-interference ability, and good performance in extreme weather environments. They have become one of the main sensors of driverless cars.

In reference [1], a tree classifier is constructed to classify obstacles by extracting features such as vehicle height, length and top length. However, due to the occlusion and self- occlusion of obstacles detected in actual road conditions, the incomplete sequence of vehicle top height affects feature extraction and vehicle recognition. In reference [2], an improved eight neighborhood region

labeling algorithm is proposed to reduce the influence of obstacle occlusion to some extent, but there is a problem of clustering two different neighboring obstacles into one obstacle. In reference [3], the method of multi single layer LiDAR sensor fusion data clustering is used to distinguish the dynamic and static attributes of obstacles, but the algorithm robustness is reduced because the use of standardized Euclidean distance in the clustering process is affected by the uneven density distribution of lidar data. In reference [4], the parameter selection of DBSCAN clustering algorithm is optimized, which overcomes the problem of uneven data density distribution when lidar detects obstacles to a certain extent, improves the accuracy of obstacle detection, but the adaptive parameter selection reduces the real-time performance of the algorithm.

In order to solve the problems of uneven distribution of data density when recognition obstacle with four-layer LiDAR, misrecognition by neighboring obstacles, and insufficient real-time performance, this paper proposes an obstacle recognition method based on four-layer LiDAR. First, a grid map is established in the area where the driver can drive, and dynamic obstacles are detected by Dempster-Shafer Theory(DST) and conflict coefficients. The grid map reduces the impact of uneven data density distribution to a certain extent. Then use the improved connected area labeling method based on segmentation to cluster obstacles. The improved connected area labeling method reduces the number of neighborhood searches and improves the real-time performance of the algorithm. In order to solve the problem of difficulty in extracting information about neighboring dynamic obstacles, a method for recognition and optimizing neighboring obstacles based on dynamic thresholds and changes in data distribution trends is proposed to more accurately identify neighboring obstacles.

2. LiDAR Data Preprocessing

In this paper, "IBEO LUX 2010" four layer LiDAR manufactured by IBEO company is used. Compared with single layer LiDAR and 64 line LiDAR, the amount of scanning data of this LiDAR is moderate, which can guarantee the feature extraction of obstacles and meet the real-time requirements of unmanned vehicles. The main performance parameters of this LiDAR are shown in Table 1.

Table 1: IBEO LUX 2010 LiDAR main parameters.

Parameter name	Parameter value
Vertical field of view	3.2°
Horizontal measurement angle	1100
Data scanning frequency	12.5/25/50HZ
Measuring range	0.3-200m
Horizontal angular resolution	0.125°
Vertical angle resolution	0.8°

In order to detect the driving area and obstacle information in front of the vehicle more comprehensively, the LiDAR is installed in the middle of the front square of the vehicle. Combined with the vehicle height information, the installation height of the LiDAR is set as 0.846m from the ground. The installation position is shown in Figure 1. Red, blue, green and yellow are used to show the four scanning layers of the four-layer LiDAR.

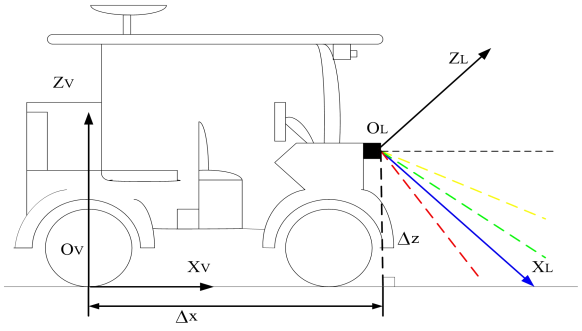

Figure 1: Schematic diagram of vehicle coordinate system.

Construction method of vehicle coordinate system in reference [5]. The x, y, z coordinates returned by the LiDAR are the coordinate values in the sensor coordinate system. Taking the blue scan layer as an example, the sensor coordinate system $X_L O_L Y_L$ and the vehicle coordinate system $X_V O_V Y_V$ are not the same coordinate system. The rotation angle of the sensor coordinate system relative to the vehicle coordinate system about the X and Z axes is 0 °, and the rotation angle about the Y axis is β. To convert the scan point $p(x, y, z)$ in the sensor coordinate system $X_L O_L Y_L$ to $p'(x', y', z')$ in the vehicle coordinate system $X_V O_V Y_V$, it needs to undergo displacement transformation and rotation transformation in the three-dimensional geometric transformation. The formula is:

$$(x', y', z') = (x, y, z)\begin{pmatrix} \cos\beta & 0 & -\sin\beta \\ 0 & 1 & 0 \\ \sin\beta & 0 & \cos\beta \end{pmatrix} + (Vx, 0, Vz) \tag{1}$$

Where Vz is the translation amount of the two coordinate systems in the z-axis direction, Vx is the translation amount of the two coordinate systems in the x-axis direction, and β is the angle between the blue scanning layer of the LiDAR and the ground.

3. Building a Grid Map of Obstacles

3.1.Build Grid Map

In order to establish a grid map, two maps should be established at the same time: one is a scan map, which is used to obtain the scan information of the current frame sensor; the other is a global map, which is used to store the scan data of the previous frame grid map and define the grid state of the unknown area. First, the global grid map is initialized to an unknown state. When the t-frame scan data is obtained, the t-scan map grid state is updated by the sensor model and fused with the global map at frame t-1. The specific establishment process of grid map is shown in Figure 2.

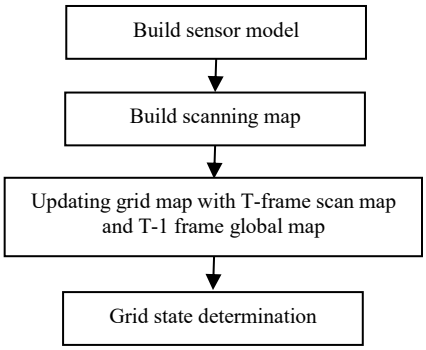

Figure 2: Flow chart of building grid map.

3.2. Dempster-Shafer Theory (DST)

DS theory is a kind of inexact reasoning theory developed in 1976. It belongs to the category of artificial intelligence. It was first used in expert system and has the ability to deal with uncertain information. As a kind of uncertain reasoning method, the main characteristics of evidence theory are: satisfying weaker conditions than Bayesian probability theory; having the ability to express "uncertain" and "unknown" directly. Based on D-S evidence theory (DST), reference [6] Used radar to detect building targets, reference [7] fused multi LiDAR information to identify targets.

The discernment framework of DST is described in reference [8]. DST discernment framework Θ consists of n exclusive elements θ_i ($i=1,2,...,n$), $\Theta=\{\theta_1,\theta_2,...,\theta_n\}$ .Mutual exclusion of elements in DST discernment framework θ_i ($i=1,2,...,n$), $\theta_k \bigcap \theta_i = \varnothing (k \neq i; k,i=1,2,...,n)$. The power set 2^Θ of discernment framework Θ is defined as follows:

- All events including empty sets Φ and Θ;
- $\theta_1 \bigcup \theta_2 \in 2^\Theta$ if $\theta_1,\theta_2 \in 2^\Theta$

In addition, DST also introduces the basic probability distribution function to solve the uncertainty problem, For event $\theta_i (i=1,2,...,n)$ in discernment framework Θ , if function m satisfies:

$$\begin{cases} m(\Phi)=0 \\ \sum_{i=1}^{n} m(\theta_i)=1 \end{cases} \tag{2}$$

then M is regarded as the basic probability distribution function of power set 2^Θ.

If X_1 and X_2 are two evidences with different properties in the discernment framework, and their basic probability functions are $m_1(X_1)$ and $m_2(X_2)$, then the DST fusion rule is

$$m(Y)=\begin{cases} 0, Y=\Phi \\ \sum_{X_1 \bigcap X_2=Y} \frac{m_1(X_1)m_2(X_2)}{1-K}, Y \neq \Phi \end{cases} \tag{3}$$

$$K=\sum_{x_1 \bigcap x_2=\Phi} m_1(X_1)m_2(X_2)<1 \ , \ K \in [0,1] \tag{4}$$

The closer the conflict coefficient K is to 1, the more inconsistent the two evidences are; on the contrary, the closer the K is to 0, the more consistent the two evidences are.

3.3.Build Scanning Map

In this paper, DST is used to build grid map. In grid map, the state of each grid unit can be barrier free state (F), obstructed (O) and the unknown state of grid is defined as $\Omega=\{F,O\}$, so the recognition framework of grid is $2^{\Omega}=\{F,O,\Omega,\Phi\}$, and the corresponding basic probability function is $[m(F)m(O)m(\Omega)m(\Phi)]$, which respectively represents the barrier free, obstructed, unknown and conflict states of a grid. Four basic probability functions satisfy $\sum_{A\subseteq\Omega} m(A)=1$.

As shown in Figure 3, The detection range of LiDAR is divided into $m\times n$ grid cells with the size of $l\times l$, where P_1 and P_2 are data points scanned by LiDAR. C_1, C_2 and C_3 are grid cells, (θ^-,θ^+) represents the maximum and minimum angle of each grid relative to the X axis. R represents the distance from the grid center point to LiDAR.

In the fusion of scanning map and global map, the coordinate value (X_c,Y_c,Z_c) of grid cell C in sensor coordinate is required, where the coordinate (X_c,Y_c) is the coordinate value of grid center point in sensor XOY plane, Z_c is the maximum height of k scanning points contained in each grid, $Z_c=\{\max(z_1,z_2,\ldots,z_k)\}$. Project n data points scanned by LiDAR into $m\times n$ grids, where x,y,z is the position information of scanning point in LiDAR coordinate system, and d represents the distance from scanning point to LiDAR center point. The point set P of grid C state is $P=\{\arctan(\frac{y_i}{x_i})\in(\theta^-,\theta^+),i\in[1,N]\}$.

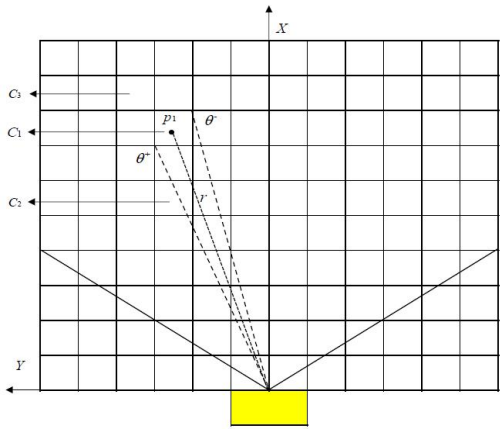

Figure 3: Grid state diagram.

The state of each grid cell C is determined by the following rules:

- If $\forall i\in[1,N],\exists d_i\in[r-\sqrt{2}l/2,r+\sqrt{2}l/2]$ and $z_c\geq 0.1\text{m}$ then the state of the grid is obstructed. For example, the grid cell C_1 in Figure 6. The basic probability function value of obstacle state is: $m(F)=0,m(O)=1-\lambda_1,m(\Omega)=\lambda_1$.

- If $\forall i \in [1,N],\ \exists d_i \in [-\infty,\ r-\sqrt{2}l/2) \cup (r+\sqrt{2}l/2,+\infty]$ and $(r+\sqrt{2}l/2) < \min(d_i)$ then the status of grid C is barrier free state. For example, the grid cell C_2 in Figure 6. The basic probability function value of barrier free state is $m(F)=1-\lambda_2$, $m(O)=0$, $m(\Omega)=\lambda_2$.
- If $\forall i \in [1,N], \exists d_i \in [r-\sqrt{2}l/2, r+\sqrt{2}l/2]$ and $(r+\sqrt{2}l/2) > \min(d_i)$ then the status of the grid is unknown. For example, the grid cell C_3 in Figure 6. The basic probability function value of unknown is $m(F)=0$, $m(O)=0$, $m(\Omega)=1$.

λ_1 and $\lambda_2 \in [0,1]$ in the rules are the false positives and missed rate of LiDAR respectively.

3.4.Update of Grid Map

The XOY grid map divides the $80m \times 32m$ area in front of the unmanned vehicle into 400×160 grid cells with a length and width of $0.2m \times 0.2m$. ε is the angle between the vehicle body of the t frame and $t-1$ frame, v is the speed of vehicle, grid $C_i^{'}$ is an estimate of the position of grid C_i at time t-1 after a scan period of $0.08s$ at time t, each $C_i^{'}$ grid in the $X'O'Y'$ grid map and a grid of t frames One-to-one correspondence of grid C_i in the map, where

$$\begin{cases} x'_{t-1} = x_{t-1} + 0.08v\cos\varepsilon \\ y'_{t-1} = y_{t-1} + 0.08v\cos\varepsilon \end{cases} \tag{5}$$

In the formula, $\varepsilon = \varepsilon_2 - \varepsilon_1$, ε_1 is the angle between the vehicle body and the road edge at frame $t-1$, ε_2 is the angle between the vehicle body and the road edge at frame t.

If $x'_{t-1} > 80m$ or $y'_{t-1} \notin (-16m,16m)$ that is, the grid C_i of $t-1$ frame moves to the unknown domain in the global map at t frame, the grid $C_i^{'}$ of t frame corresponding to C_i is fused with the grid of unknown state; if $x'_{t-1} \le 80m$ and $y'_{t-1} \in (-16m,16m)$ that is, the grid C_i of $t-1$ frame moves to the known domain in the global map at t frame, the grid C_i of t frame corresponding to $C_i^{'}$ and the grid C_j closest to C_i of $t-1$ frame are fused.

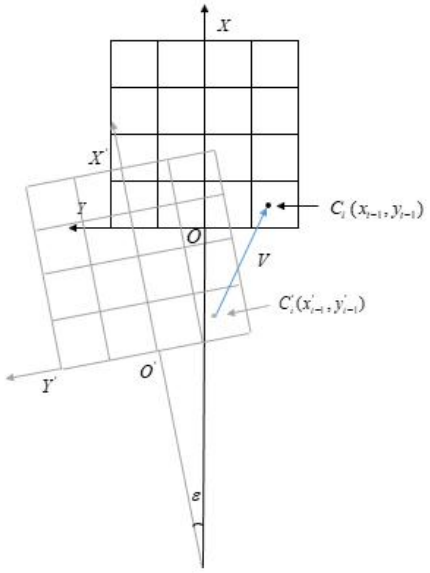

Figure 4: Position estimation of local grid map.

Assuming that the basic probability function of each grid of the scanned map at time t is m_1, and the basic probability function of each grid of the global map at time $t-1$ is m_2, the DST fusion formula is

$$\begin{cases} m(\phi)=0 \\ m(F)=\dfrac{m_1(F)m_2(F)+m_1(F)m_2(\Omega)+m_1(\Omega)m_2(F)}{1-K} \\ m(O)=\dfrac{m_1(O)m_2(O)+m_1(O)m_2(\Omega)+m_1(\Omega)m_2(O)}{1-K} \\ m(\Omega)=1-m(F)-m(O) \end{cases} \qquad K = m_1(F)m_2(O)+m_1(O)m_2(F) \tag{6}$$

4. Dynamic Obstacle Recognition

4.1.Dynamic Obstacle Detection

In this paper, the collision coefficient K in fusion rules is used to detect dynamic obstacles. Reference [9] divides the conflict coefficient into two parts according to the obstacle state: $K = C_1 + C_2 = m_1(F)m_2(O) + m_1(O)m_2(F)$ In the formula, $C_1 = m_1(F)m_2(O)$ is a grid from the barrier free state at $t-1$ time to the obstructed state at t time, that is, dynamic obstacles enter the grid; $C_2 = m_1(O)m_2(F)$ means that a grid changes from an obstructed state at any time to a barrier free state at any time, that is, the dynamic obstacle leaves from the grid.

4.2.Expansion Erosion Treatment

Because LiDAR has a certain rate of missed detection and false alarm, or because of the influence of the shape and weather of obstacles, when using LiDAR to obtain the data points of obstacles, the scanning points of the same obstacles may break into multiple obstacles, which greatly interferes with the clustering and information extraction of obstacles. Therefore, it is necessary to use the

expansion erosion algorithm to connect the fracture of the same obstacle and lubricate the contour of the obstacle.

4.3.Obstacle Clustering based on Segment Markers

After the dynamic obstacles are treated by expansion erosion, a continuous and smooth obstacle can be obtained. Next, we will cluster the dynamic obstacles. The significance of clustering is to summarize the characteristics of the same dynamic obstacles and make further analysis, such as the size of obstacles and the distance between obstacles and driverless vehicles.

The classical marking algorithm based on region growing technology scans the map from bottom to top, from left to right. When an unmarked obstacle grid is encountered, the region marking algorithm is executed, and then continues to scan backward. The execution process of the classical region marking algorithm is: take the current obstacle grid as the seed, mark the current grid with a new label, then mark the adjacent obstacle grid with it, then mark the neighborhood obstacle grid with its adjacent grid as the seed, and repeat the process until the neighborhood grid of all seeds has become the marked state.

Reference [10] analyzes the disadvantage of this classical region marking algorithm is that the connected grids between the target points are searched repeatedly, resulting in a large number of redundant neighborhood searches. Reducing the number of neighborhood searches is an effective way to improve the efficiency of this algorithm.

Reference [11] proposes to detect diseased paper by processing image pixels through segment-based connected area labeling. In this paper, we propose an improved connected region labeling method based on segment connected region labeling. Segment refers to the continuous grid projected by an obstacle point cloud in the scan line. As shown in Figure 5 (any grid can be barrier grid or barrier free grid). The grid between grid s and grid e is called segment, which belongs to a connected domain. The process of segment labeling reduces the number of times that connected grids are searched repeatedly in traditional methods, and improves the real-time performance of the algorithm

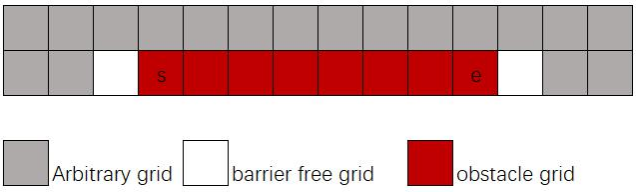

Figure 5: Schematic diagram of segment-based connected region labeling.

Segment based connected area labeling algorithm scans the map from bottom to top and from left to right. When the left neighborhood of an unmarked obstacle grid is an obstacle free grid, the obstacle grid is the segment head grid of the current segment, which is recorded as GS, and its location is stored in the segment head array. When it is detected that the right neighborhood of an obstacle grid in a segment is an obstacle free grid, the current obstacle grid is the segment tail grid, which is recorded as GE, and its position is stored in the segment tail array. According to whether the current segment has adjacent segments in the previous line, the current segment is marked in two cases:

a) There is no adjacent segment in the previous row of the current segment, indicating that the current segment is the first row of the newly emerging obstacle area. Use the new label to mark the segment, and then add one to the corresponding label variable value.

b) There is an adjacent segment in the previous row of the current segment:

- When $i=1$, mark the current segment with the label of the adjacent segment;

- When $i>1$, the minimum label is selected as the label of the current segment from multiple adjacent segments, and the equivalent combination is performed.

4.4. Neighboring Obstacles

In the complex traffic environment, when the number of vehicles increases and the distance between vehicles is relatively close, the accuracy of LiDAR obstacle clustering algorithm will inevitably decline, which is reflected in the easy recognition of neighboring obstacles as an obstacle.

The distance between the nearest two points is less than the distance threshold, then it is regarded as the neighboring obstacles.

If there are z connected regions and g obstacle grids in each connected region, then:

- Starting from the connected area labeled 1, the distance between g grid center points in labeled 1 and other $z-1$ connected areas is calculated in turn. Record the nearest grid pair, which is recorded as the adjacent grid.
- Loop until all connected areas record the adjacent grid.
- Starting from the connected area labeled 1, the distance between the points in each group of adjacent grids in labeled 1 is calculated in turn. The adjacent grid with the distance less than the distance threshold is recorded, and the connected area of the obstacles where the two grids are located is recorded as the neighboring obstacles.
- Loop until all adjacent grids are calculated.

4.5.Adaptive Distance Threshold

Borges in reference [12] proposed a method to determine the distance threshold D based on the depth value

$$D = r_{n-1}\frac{\sin(\Delta\phi)}{\sin(\lambda-\Delta\phi)}+3\sigma_r \tag{7}$$

As shown in Figure 6, r_{n-1} is the depth value of point p_{n-1}, σ_r is the measurement error of LiDAR, $\Delta\phi$ is the angular resolution of LiDAR, and λ is the threshold parameter. The value of λ determines the size of threshold D. the larger λ is, the smaller D is. In the algorithm of vehicle target recognition, the distance between successive points on the surface of obstacles with a certain distance is calculated, and λ value is deduced according to the formula.

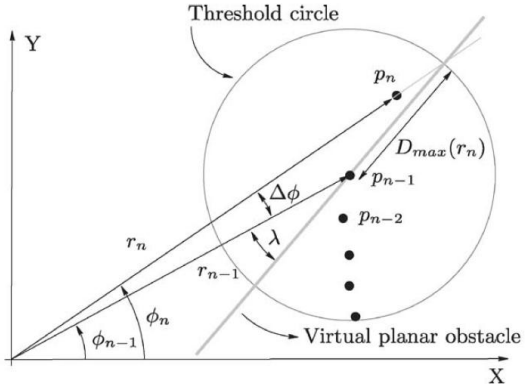

Figure 6: Borges threshold diagram.

Reference [13] proposed the characteristics of vehicle targets scanned by LiDAR, the density of data points on two sides of the vehicle is relatively different, and the density of data points on the front of the vehicle is relatively small. There are two scenarios when an adjacent obstacle is identified as an obstacle:

a) As shown in Figure 7, two vehicles are identified as an obstacle in the forward direction because the front of the front vehicle is adjacent to the obstacle grid on the side of the rear vehicle.

b) As shown in Figure 8, two vehicles in the forward direction are identified as one obstacle due to the adjacent obstacle grid on the side of the vehicle.

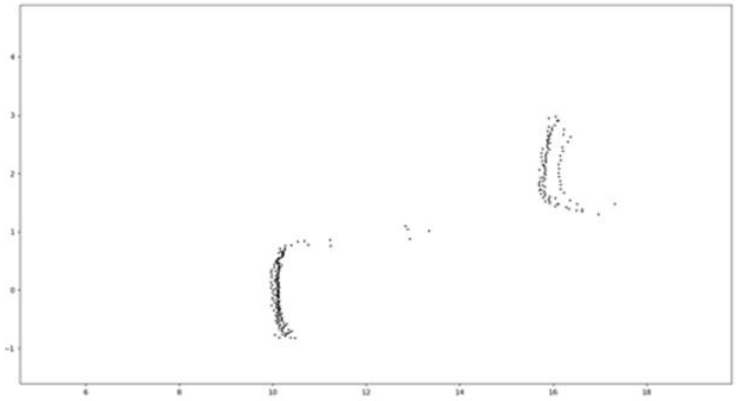

Figure 7: Front and rear occluded neighboring obstacles.

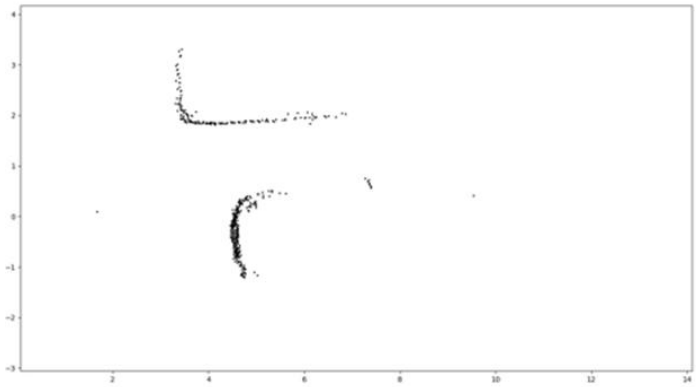

Figure 8: Left and right occluded neighboring obstacles.

In order to make the distance threshold better adapt to the vehicle model. First, calculate the slope of the line between two adjacent points relative to the x-axis direction. If the slope is small, the two adjacent points are likely to be on the front of the two vehicles, and the larger the threshold factor λ is, the smaller the distance threshold is; if the slope is large, the two adjacent points are likely to be on the side of the vehicle, and the smaller the threshold factor λ is, the larger the distance threshold is.

The specific calculation process is as follows:

- Calculate the absolute value ratio of the difference between the abscissa and the ordinate of p_{n-1} and p_n points $a = \left| \frac{x_n - x_{n-1}}{y_n - y_{n-1}} \right|$, where (x_n, y_n) represents the coordinate of p_n point and (x_{n-1}, y_{n-1}) represents the coordinate of p_{n-1} point.

- If $a < A$, point p_{n-1} and point p_n may be in the same vertical plane, λ is the smaller value, if $a > A$, λ is the larger value; calculate the value of distance threshold D according to formula 7. Where A is the threshold value of a value.

4.6. Dynamic Adjacent Obstacle Recognition

As shown in Figure 9 and Figure 10, in the process of moving nearby obstacles, due to the phenomenon that LiDAR scanning is blocked, part of the point cloud data will be lost, and the loss point cloud is mainly concentrated on the side of the vehicle. Through the analysis of the loss of point cloud in three directions, as shown in Figure 8, it is found that the distribution of point cloud in the axial direction before and after the loss of point cloud can be used as the basis for the identification and optimization of neighboring obstacles. The direction point cloud is divided into (0-0.2), (0.2-0.4), (0.4-0.6), (0.6-0.8), (0.8-1.0) five intervals from small to large.

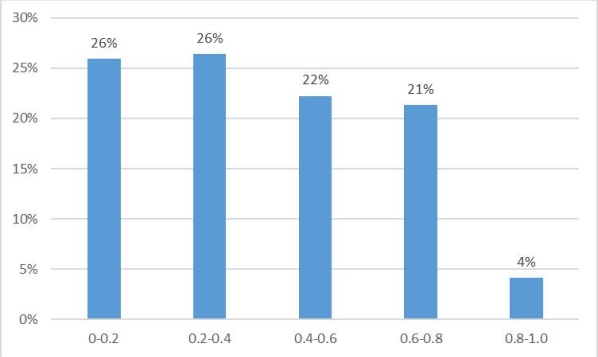

Figure 9: The distribution of z-axis data before false recognition of neighboring obstacles.

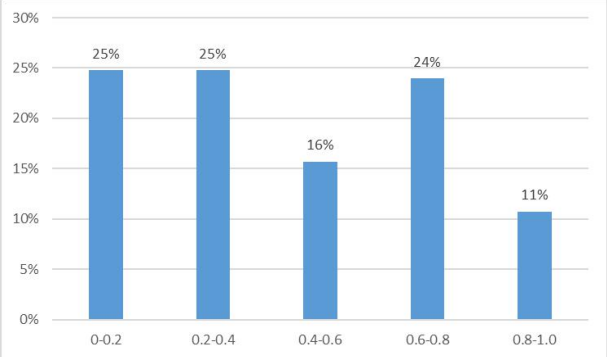

Figure 10: The distribution of z-axis data after the recognition of neighboring obstacles.

When the neighboring obstacles are identified as an obstacle, the data points in the 0.4-0.6 range in the z-axis direction show a significant downward trend, and the data points in the 0.8-1.0 range show a significant upward trend.

According to the rule of the change of z-axis data points in the process of neighboring obstacles being mistakenly identified, the recognition of neighboring obstacles is optimized.

- When the two neighboring obstacles at the k time disappear at the next time, the two neighboring obstacles at the k time are paired with the nearest obstacle block i in the connected area result at the next time to obtain a matching obstacle list [13]. The following parameters of the obstacle are mainly used: the coordinates of the center of the obstacle, the direction of movement,

the speed, and the mean value of the intensity. (x_k^i, y_k^i) and T_k^i are the center coordinates of the i obstacle and the mean value of the scanning point intensity at time k ; t is the time difference between the two frames of data; v_i is the speed of the obstacle; θ_i is the direction of the obstacle's movement; T_i is the two matching obstacle The average value of the laser scanning point intensity of the object. The calculation formula is

$$\begin{cases} v_i = \sqrt{(y_{k+1}^i - y_k^i)^2 + (x_{k+1}^i - x_k^i)^2} / t \\ \theta_i = \arctan\left(\left| \dfrac{y_{k+1}^i - y_k^i}{x_{k+1}^i - x_k^i} \right| \right) \\ T_i = (T_{k+1}^i + T_k^i) / 2 \end{cases} \tag{8}$$

- According to the obstacle matching method proposed in reference [14]. After obtaining the obstacle block information, the similarity calculation is performed on the two obstacle blocks at the k time and the obstacle blocks at the $k+1$ time. a, b, c, d are weight values, and the weight value and the similarity threshold are obtained by experimental fitting.

$$s_{ij} = \frac{a}{(x_i - x_j)^2 + (y_i - y_j)^2 + 1} + \frac{b}{(v_i - v_j)^2 + 1} + \frac{c}{(\theta_i - \theta_j)^2 + 1} + \frac{d}{(T_i - T_j)^2 + 1} \tag{9}$$

- When the spatial similarity between two neighboring obstacles at time k and the obstacle at time $k+1$ is less than the similarity threshold, the change trend of the point cloud distribution in the z-axis direction is calculated.
- When the point cloud distribution on the z-axis shows a downward trend in the point cloud in the 0.4-0.6 interval and an upward trend in the 0.8-1.0 interval, the obstacle block at the moment is judged as the neighboring obstacle at the moment and marked.

The preferred spelling of the word “acknowledgment” in America is without an “e” after the “g”. Avoid the stilted expression “one of us (R. B. G.) thanks ...”. Instead, try “R. B. G. thanks...”. Put sponsor acknowledgments in the unnumbered footnote on the first page.

5. Analysis of Experimental Results

The hardware platform used in the experiment was the Intel Core i5-7400 CPU 3GHz, the memory was 8GB, and the operating system was Microsoft Windows 10.

5.1. The Establishment of Grid Map

Figure 11 is the experimental scene, Figure 12 is the LiDAR scanning point after the ground point is filtered according to the height, Figure 13 is the grid map fused according to DST, and the false alarm rate λ_1 and missed detection rate λ_2 of LiDAR are set to 0.1;

Figure 11: Experimental scene.

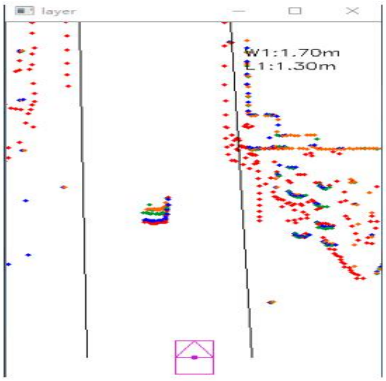

Figure 12: LiDAR point cloud

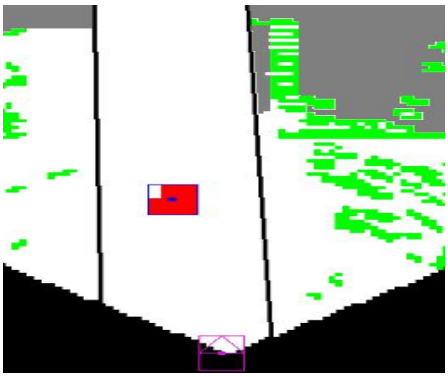

Figure 13: Grid map.

5.2.Obstacle Clustering and Analysis

This paper compares the segment-based connected region labeling algorithm with a fixed threshold eight-neighborhood labeling algorithm. Through the analysis of 480 obstacle experiments, the accuracy rate of obstacle clustering is shown in Table 2, and the algorithm execution speed is shown in Table 3. As shown. The algorithm in this paper has better clustering effect within 30m of obstacle distance. The clustering effect of the eight-connected labeling algorithm decreases significantly with the increase of obstacle distance. Compared with the eight-connected labeling algorithm, the algorithm execution time in this paper is significantly reduced, and the algorithm execution time is relatively more stable when the number of obstacles increases.

Table 2: Comparison of accuracy of clustering algorithms.

Obstacle distance/m	Eight-connected labeling algorithm accuracy/%	Algorithm accuracy of this paper /%
0-10	91.2	92.3
10-20	87.9	86.6
20-30	55.8	72.9
30-40	20.7	66.1

Table 3: Comparison of time performance of clustering algorithms.

Number of obstacles	Eight-connected labeling algorithm execution time/ms	Algorithm execution time of this paper/ms
1	58	36
2	72	51
3	114	59
4	170	68

5.3.Neighboring Obstacle Recognition Analysis

The effect of recognition neighboring obstacles in 10 groups of different scenes was counted, and 142 samples of adjacent vehicle obstacles were analyzed. For adjacent points on the front of the vehicle, the threshold λ is set to 9 °, and for adjacent points on the side of the vehicle, the threshold λ is set to 3 °. The recognition effect is shown in Table 4. The optimized neighboring obstacle recognition has a significant improvement in 20m. The distance between neighboring obstacles is insufficient due to the reduced data density, which leads to some false detections.

Table 4: Comparison of accuracy of neighboring obstacle recognition.

Obstacle distance/m	Recognition accuracy before improvement /%	Improved recognition accuracy /%
0-10	75.3	82.6
10-20	65.4	80.4
20-30	45.3	64.7
30-40	14.1	35.8

6. Conclusions

Aiming at the problem that the traditional algorithm takes a long time to identify obstacles using four-layer LiDAR and is affected by the uneven change of data occlusion density, an improved connected region marking algorithm based on segment is proposed for grid map based on Dempster Shafer The theory of theory (DST) and the obstacles detected by the conflict coefficient are clustered, which reduces the search times of the traditional eight neighborhood marking method and increases the real-time performance of the algorithm. In order to solve the problem that the accuracy of four-layer LiDAR is not enough, Borges distance threshold is improved by combining the characteristics of obstacle data, and the recognition of adjacent obstacles is optimized by the improved distance threshold and the distribution characteristics of obstacle data. Experimental results show that the algorithm not only ensures the accuracy of obstacle recognition, but also improves the real-time performance of the algorithm. The algorithm improves the near distance recognition of adjacent obstacles, but with the increase of distance, the accuracy of the algorithm decreases. In the follow-up work, the obstacle recognition and tracking after the distance increase will be further studied.

References

[1] Wang Ninghan, "Vehicle recognition and classification system based on laser ranging," [D]. Tianjin University, 2012.

[2] Duan Jianmin, Zheng Kaihua and Zhou Junjing, "Environment perception of multi-layer lidar in driverless vehicle,". Journal of Beijing University of Technology, 2014, 40(12): 1891-1898.

[3] Bao Kan, "Research on near field object detection and state recognition of intelligent vehicle" [D]. Jilin University, 2016.

[4] Duan Jianmin, Ren Lu, Wang Changren and Liu Dan, "Road information extraction and target detection based on four line lidar," Laser Journal, 2017, 38(06): 32-37.

[5] Duan Jianmin, Zheng Kaihua, Li Longjie and Shi Xiaoli, "Road information extraction algorithm based on multi-layer lidar" [J]. control engineering, 2016, 23(04): 468-47.

[6] Zhao Qi, "SAR building target detection based on morphology and DS evidence theory," [D]. Xidian University, 2017.

[7] Han Yujie, Wang Guohong and Wang Na, "A multi radar target recognition method based on DS evidence theory"[J]. Journal of missiles and guidance, 2009,29(05): 215-220.

[8] Yuan Shuai, Guo YanRu, Gong Wei, Han Xiaoying and Yan Xue, "Study on the model of indoor environment contour ultrasonic detection based on DSmT" [J]. Journal of instrumentation, 2018, 39.

[9] MORAS Julien， CHERFAOUI Véronique, BONNIFAIT Philippe． Credibilist occupancy grids for vehicle perception in dynamic environments[C]//2011 IEEE International Conference on Robotics and Automation． Shanghai: IEEE, 2011: 84-89．

[10] Guo Feiyan, Li Xiaojing, "Application of improved connected region marking algorithm in facial features recognition and location technology" [J]. Science and Technology Bulletin, 2019, 35(05): 71-74.

[11] Zhao Xiao, "Research on high speed of connected region marking algorithm and its application in paper defect detection" [D]. Shaanxi University of science and technology, 2019.

[12] BORGES G， ALDONM． "Line extraction in 2D range images for mobile robotics"[J]. Journal of Intelligent and Robotic Systems, 2004, 40: 267-297.

[13] Zhou Junjing, "Research on Key Technologies of target recognition and tracking of intelligent vehicle based on lidar" [D] Beijing University of Technology, 2014.

[14] Lou Xinyu, Wang Hai, Cai Yingfeng, Zheng Zhengyang and Chen Long, "Research on real-time road obstacle detection and classification algorithm using 64 line lidar"[J]. Automobile Engineering, 2019,41(07): 779-784.