Proceedings of the 34th Chinese Control Conference
July 28-30, 2015, Hangzhou, China

Road and Obstacle Detection Based on Multi-layer Laser Radar in Driverless Car

DUAN Jianmin[1], ZHENG Kaihua[1],SHI Lixiao[1]

1. Academy of Control Science and Engineering, Beijing University of Technology, Beijing 100124
E-mail: 1034746792@qq.com

Abstract: To make a driverless car with better environment awareness, multi-layer laser radar was applied to detect roads and obstacles. Firstly the road edge data set was extracted from numerous laser radar data based on characteristics of the road edge data, and the cluster analysis of the data sets was done with the improved COBWEB algorithm based on Euclidean distance. In order to divide the road into drivable area and undrivable area, the left and right road edges were respectively fitted into a straight line with the least squares method. Secondly DSmT was applied to establish a grid map for the environment, and dynamic obstacles were detected by the conflict coefficient within drivable area. Finally, the cluster analysis and information extraction of dynamic obstacles was completed by the expansion algorithm, erosion algorithm and improved eight neighborhood labeling algorithm. The results show that the algorithm can significantly reduce redundant operations and improve efficiency.

Key Words: Multi-layer laser radar, Driverless car, Euclidean distance, COBWEB algorithm, DSmT, Grid map

1 Introduction

The driverless cars in an urban environment need a good perception of the surroundings, including the perception of road structure and the detection of other obstructions and so on. Reliable environmental awareness plays a crucial role in autonomous cruise control, collision warning and path planning.

Usually camera, radar, GPS and other sensors can be installed in unmanned vehicle to perceive the environment. Laser radar has excellent features of being not affected by weather and illumination. In addition, the laser radar has the advantages of high scanning frequency, rich data volume and facilitating rapid processing. Therefore, the use of laser radar has better robustness and rapidity to perceive environment information around the driverless car.

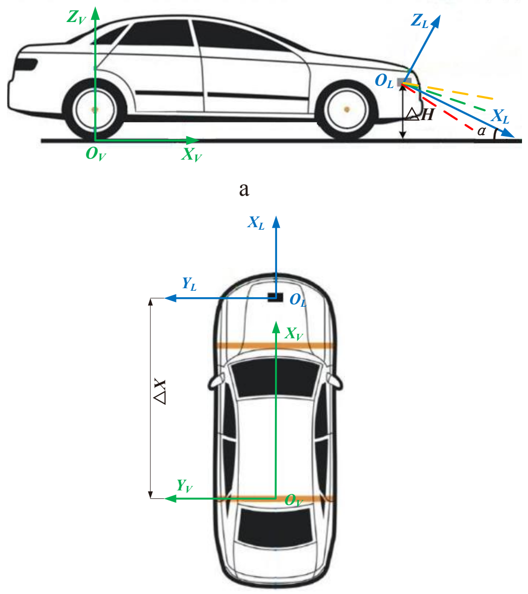
2 Lidar Performance and Coordinate System

Laser radar can be classified into single-line, four-line and 64-line laser radar by the number of scanning lines. Compared to single-line laser radar, four-line laser radar has the advantages of abundant data, high precision, far detecting distance and wide range [1]. Although four-line laser radar is not as good as 64 -line in detection accuracy, range and distance, four-line laser radar can better meet the real-time requirements of unmanned vehicles due to its moderate amount of data. We chose IBEO-LUX 2010 four-line laser radar as a sensor, which can accurately and steadily acquire the road edge information and detect dynamic obstacles. Performance parameter of IBEO_LUX is shown in Table 1.

Table 1: Performance Parameter of IBEO_LUX

Performance	Parameter
Horizontal view angle	85°（-50°~35°）
Vertical view angle	3.2°
Scanning frequency	12.5/25.0/50.0 Hz
Angular resolution	Hor：0.125°Ver：0.8°
Range resolution	4cm
Measurement range	0.3 ~ 200 m

The car and sensor coordinate system is shown in Fig.1. Red, blue, green and yellow represent the four scanning layers of laser radar from bottom to top. Additionally, the laser radar is inclined downward and α represents depression angle.



a

b

Fig.1: Establish the coordinate system (a: lateral view, b: vertical view)

3 Road Edge Detection Algorithm

The main role of detecting road edge is to divide the road into drivable area and undrivable area. The focus of processing algorithm is limited to drivable area by road segmentation, which not only reduces the amount of data processing, but also reduces the interference information [2][3].

3.1 Extraction of Road Edge Data Sets

Through experimental analysis, summarize and conclude the characteristics of the road edge data: when the laser radar scans on road edge, data points returned in the same scan layer exhibits stable continuity of serial number, and these successive road edge data points can be described by the linear function $y = kx + b$, where k is the slope, b is

*This work is supported by Beijing Natural Science Foundation (BNSF) of China under Grant JJ002790200802.

the intercept. We utilize these features to extract road edge data set from numerous radar scan points. The flow chart of road edge data set extraction algorithm is shown in Fig.2.

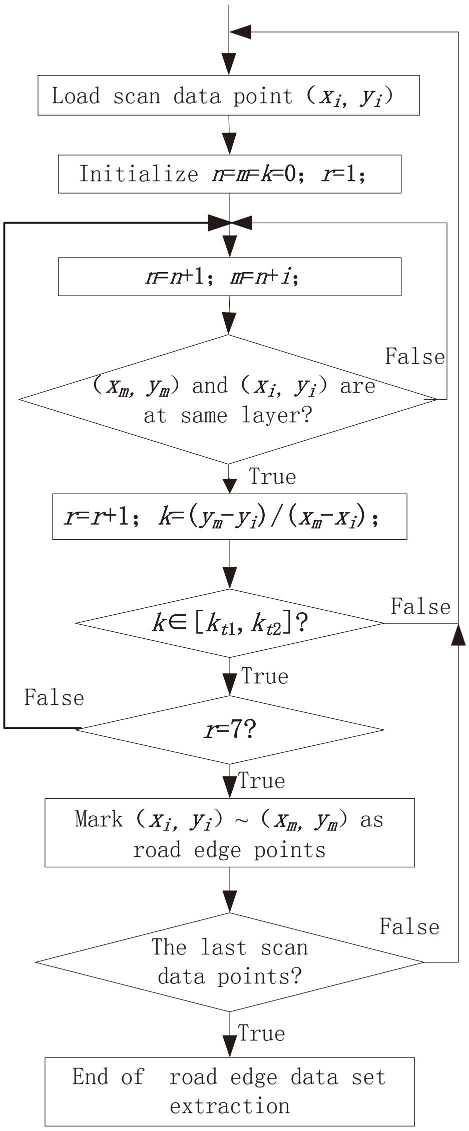

Load scan data point (x_i, y_i)
Initialize n=m=k=0; r=1;
n=n+1; m=n+i;
(x_m, y_m) and (x_i, y_i) are at same layer?
False
True
r=r+1; k=(y_m−y_i)/(x_m−x_i);
k∈[k_t1, k_t2]?
False
True
False
r=7?
True
Mark (x_i, y_i) ~ (x_m, y_m) as road edge points
The last scan data points?
False
True
End of road edge data set extraction

Fig.2: Flow chart of road edge data set extraction algorithm

In Fig.2, i is the ordinal number of current data point; n is the increment of ordinal number; m is an ordinal number corresponding to the （$i+n$）th data point; k is a slope between road edge data points; $[k_{t1}, k_{t2}]$ is variable threshold range of slope between road edge data points, $k_{t1}=\lambda-\varepsilon$, $k_{t2}=\lambda+\varepsilon$, $\lambda=\tan\beta$, β is the angle between lane line and vehicle driving direction; ε is measuring error; r is the number of data points accord with the characteristics of the road edge data, empirical value is 7 .

3.2 The Improved COBWEB Algorithm Based on Euclidean Distance

In this paper, we apply the Euclidean distance to improve COBWEB clustering algorithm, and then use improved COBWEB algorithm to cluster the road edge data set.

Given the model characteristics of road edge data set, four attributes are created for each road edge point: The slope k_n and intercept b_n are created between the current road edge point and the next road edge point. The slope k_l and intercept b_l are created between the current road edge point and the last road edge point. After creating a four-dimensional attribute space, Euclidean distance can be calculated in this space. The greater distance, the lower similarity of two road edge points. The smaller distance, the higher similarity. Euclidean distance between P_i and P_j is calculated as (1)

$$d=\sqrt{\begin{array}{l}(P_i.k_n-P_j.k_n)^2+(P_i.b_n-P_j.b_n)^2\\+(P_i.k_l-P_j.k_l)^2+(P_i.b_l-P_j.b_l)^2\end{array}} \tag{1}$$

The COBWEB algorithm uses the classification tree to cluster. Each node of the tree is the description of data attribute information. COBWEB algorithm uses CU (Category Utility) to guide the establishment of tree. The procedure is, temporarily place the object in each node, and calculate the CU value, make the appropriate action based on CU value [4]. The COBWEB algorithm is able to dynamically adjust the number of clusters.

In this paper, the Euclidean distance is used to improve the COBWEB algorithm. The improved COBWEB algorithm is contributed to the cluster analysis of road edge data set. The improved CU is defined as (2)

$$CU_e=\frac{\sum_{k=1}^{n}p(C_k)[\sum_i\sum_j p(d<\zeta\mid C_k)^2-\sum_i\sum_j p(d<\zeta)^2]}{n} \tag{2}$$

In equation (2), n is the number of clusters, C_k is the k th cluster, $p(C_k)$ is the ratio between the number of road edge in C_k and the whole set, ζ is the threshold value of Euclidean distance, $p(d<\zeta\mid C_k)$ is the probability of $d<\zeta$ between P_i and the other road edge point in C_k , $p(d<\zeta)$ is the probability of $d<\zeta$ between P_i and the whole road edge data set.

The step of improved COBWEB algorithm is as followed [4]:

（1） Road edge data set（$P_1, P_2 \ldots P_n$）obtained from the pre-processing stage，and add P_i to the clustering algorithm according to the serial number from small to large order.

（2） Create a classification tree and initialize.

（3） P_i is temporarily placed in each node, and calculate CU_e. Looking for a node that is similar with P_i according to CU_e .If P_i does not belong to any node，create a new node.

（4） Whether to merge or split operations，the fifth step is executed to merge， the sixth step is to split.

（5） Merge：according to CU_e， select the highest score and the second highest score of the nodes to merge.

（6） Split：according to CU_e， select the highest score node to split.

（7） Add P_{i+1} to the clustering algorithm until the last road edge point is done.

3.3 The Experimental Results of Road Edge Detection

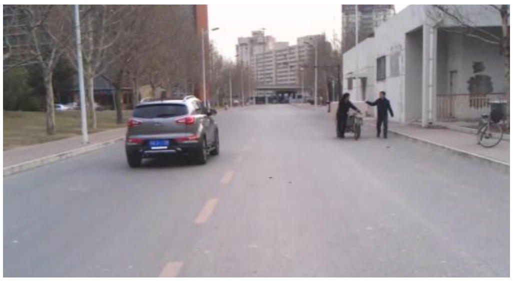
Experimental platform is BJUT-IV unmanned vehicle. IBEO-LUX four-line laser radar, GPS, INS and camera is installed on BJUT-IV. Experimental environment is within BJUT campus road, average speed of BJUT-IV is 10km / h. The scanning frequency of radar is set to 12.5Hz, measurement error $\varepsilon = 0.04$, and radar was installed at a height of 0.846 meter from the ground. Depression angle of the radar $\alpha = 1.6^\circ$, which can ensure that road edge can always be detected in the red and blue layers, green and yellow layers are focused on obstacles detection. The scan data of Laser radar will be transmitted to processing algorithm by 100MBit / s Ethernet.

In Fig.3.a, the angle between travel direction and lane $\beta = 1.3^\circ$, variable threshold range of slope between road edge data points $[k_{t1}, k_{t2}] = [-0.017, 0.063]$.Laser radar raw data map is shown in Fig.3.b.

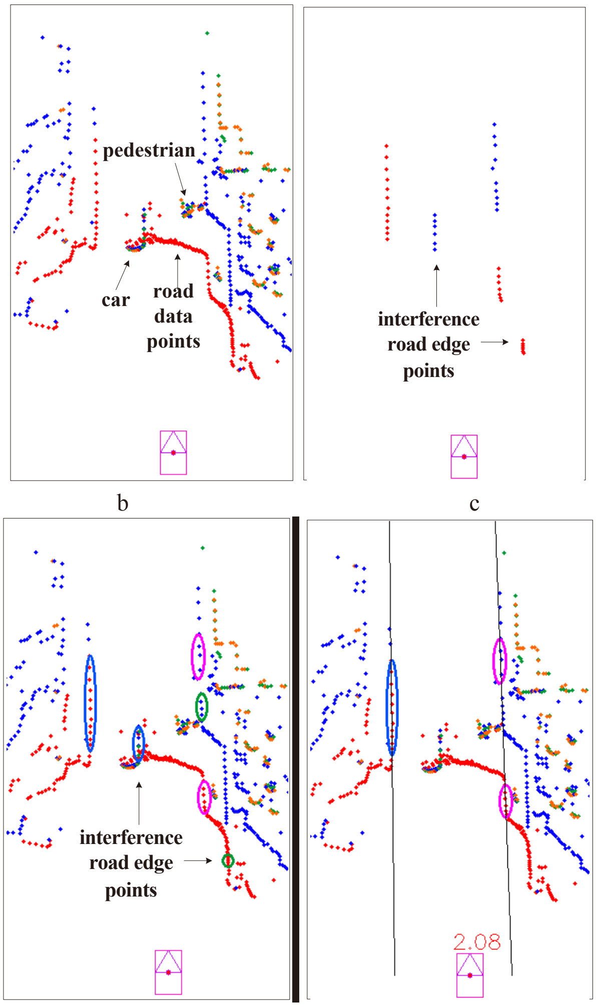
In Fig.3.c, the road edge data set were extracted from numerous laser radar data according to characteristics of the road edge data. The set includes real road edge point, also includes interference road edge point. For example, blue interference road edge point caused by scanning to the vehicle body and the red interference point caused by scanning to the wall.

In Fig.3.d, the result of cluster by using the improved COBWEB algorithm based on Euclidean distance. The oval automatically generated after clustering.

In Fig.3.e, work is to divide categories into the left and right road edge, and remove interference road edge. The left and right road edge was respectively fitted a straight line with the least squares method [5]. The angle between travel direction and right road edge $\gamma = 2.08^\circ$ is displayed above the unmanned vehicle.



a



b

c

d

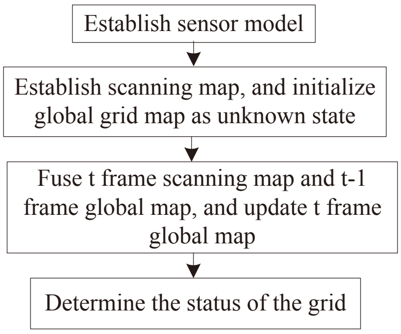
e

Fig.3: Experimental results of road edge detection

4 Grid Map Establishment and Dynamic Obstacles Detection

In this paper, DSmT is applied to establish grid map to depict road environment, vehicles, pedestrians and other information. When establishing grid map, since Dempster-Shafer theory will cause counterintuitive results in some cases [6]. DSmT is developed on the basis of DST, and can be a good solution to above problem.

4.1 Grid Map Establishment



Establish sensor model
Establish scanning map, and initialize global grid map as unknown state
Fuse t frame scanning map and t-1 frame global map, and update t frame global map
Determine the status of the grid

Fig.4: Flow chart of grid map

For the establishment of a grid map, should create two maps: one is scanning map, its role is to obtain the current scan frame information on laser radar. The other is global map, whose role is to store the data on last frame grid map and define the cell status in unknown area [7]. The specific process of establishing grid map is shown in Fig.4.

In the grid map, the status of each cell can be free (F) and occupied (O), unknown state is $\Omega = \{F, O\}$. The power set is defined as $2^{\Omega} = \{F, O, \Omega, \Phi\}$. For each cell, a mass function is calculated and provides four beliefs on the state of the cell $[m(F)m(O)m(\Omega)m(\Phi)]$, where mass functions represent respectively that the space is free, occupied, unknown or conflict. Mass functions verify the property $\sum_{A \subseteq \Omega} m(A) = 1$.

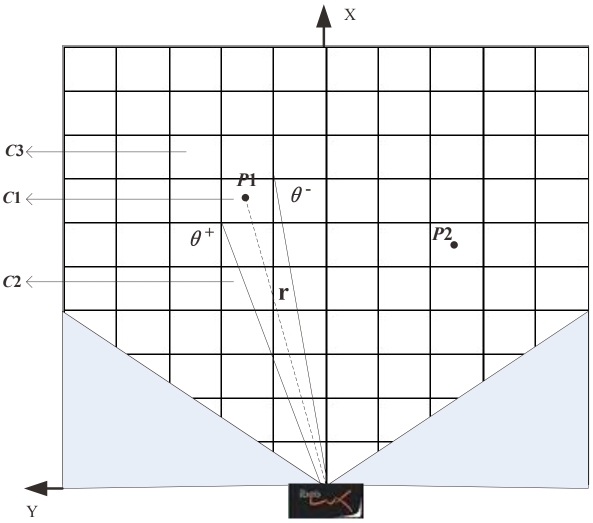


Fig.5: Sensor model (p1 and p2 are data points. C1, C2and C3are grid units)

The space is divided by $m \times n$ cells with a size of $l \times l$ as shown in Fig.5. The angel scale (θ^-, θ^+) of the cell, where, θ^- and θ^+ represent the lower and upper angle limits of the cell respect to the sensor. r is the distance from the cell center to the sensor. To fuse scanning and global map, coordinate value of each cell (x_c, y_c, z_c) is needed in the sensor coordinate, where (x_c, y_c) is center coordinate value of each grid in XOY plane, z_c is the maximum height of k scan points in each cell. N data points $p_i = \{x_i, y_i, z_i, d_i\}$ are projected to $m \times n$ grid cells, point set that can determine grid C states is defined as (3).

$$P = \{p_i . \tan^{-1}(\frac{y_i}{x_i}) \in (\theta^-, \theta^+), i \in [1, N]\} \quad (3)$$

For each cell C, the cell state is determined by the following principle:

- $\forall i \in [1, N]$, $\exists d_i \in [r - \sqrt{2}l/2, r + \sqrt{2}l/2]$, and $C.z_c > 0.1m$, then the cell state is occupied (O) , such as C1 grid cell in Fig.5. Occupied state mass functions are defined as followed: $m(F) = 0$, $m(O) = 1 - \lambda_1$, $m(\Omega) = \lambda_1$.
- $\forall i \in [1, N]$, $\nexists$ $d_i \in [r - \sqrt{2}l/2, r + \sqrt{2}l/2]$,and $(r + \sqrt{2}l/2) < \min(d_i)$, then the cell state is free (F) , such as C2 grid cell in Fig.5. Free state mass functions are defined as followed: $m(F) = 1 - \lambda_2$, $m(O) = 0$, $m(\Omega) = \lambda_2$.
- $\forall i \in [1, N]$, $\exists d_i \in [r - \sqrt{2}l/2, r + \sqrt{2}l/2]$,and $(r + \sqrt{2}l/2) > \min(d_i)$,then the cell state is unknown (Ω) , such as C3 grid cell in Fig.5. Unknown state mass functions are defined as followed: $m(F) = 0$, $m(O) = 0$, $m(\Omega) = 1$.

Where, λ_1 and $\lambda_2 \in [0,1]$ are the miss detection rate and false alarm rate separately. The cells out of the scan scale of the laser are defined to be unknown, just like the grey part in Fig.5.

Assumed that the mass function of each grid in t frame s canning map is m_1 , the mass function of each grid in t-1 fra me global map is m_2 , the DSmT fusion rule of scanning ma p and global map is defined as (5):

$$\begin{cases} m(\Phi) = 0 \\ m(F) = \dfrac{m_1(F)m_2(F) + m_1(F)m_2(\Omega) + m_1(\Omega)m_2(F)}{1 - K} \\ m(O) = \dfrac{m_1(O)m_2(O) + m_1(O)m_2(\Omega) + m_1(\Omega)m_2(O)}{1 - K} \\ m(\Omega) = 1 - m(F) - m(O) \end{cases} \quad (5)$$

Conflict coefficient $K = m_1(F)m_2(O) + m_1(O)m_2(F)$.

4.2 Dynamic Obstacles Detection

Conflict coefficient K was used to detect dynamic obstacles, which can be split into two parts [8], $K = C_1 + C_2 = m_1(F)m_2(O) + m_1(O)m_2(F)$, when $C_1 > \varepsilon_1$,which indicates that a free cell in global map is fused with an occupied cell in scanning map, and when $C_2 > \varepsilon_2$,which indicates that an occupied cell in global map is fused with a free cell in scanning map. Theoretically, the direction of dynamic obstacles can be detected by these two parts of the conflict information (from C_2 to C_1). ε_1 and ε_2 are conflict coefficient threshold.

Due to the discontinuity of the raw sensor data, sometimes the cells belonging to one object do not connect to each other, which caused great disturbance to clustering and information extraction of obstacles. We implemented dilation and erosion operation to connect the nearby cells and repair the holes in an object.

Classical eight neighborhood region labeling algorithm just needs one scan to complete the marking process, which can meet the requirements of real-time data processing.

However, classical method leads to a large number of redundant neighborhood searching. Reduce the number of neighborhood search is an effective way to improve the efficiency of the algorithm [9] [10].

After the dilation and erosion operation, obstacles would become continuous and smooth, which greatly facilitate the clustering and information extraction of obstacles.

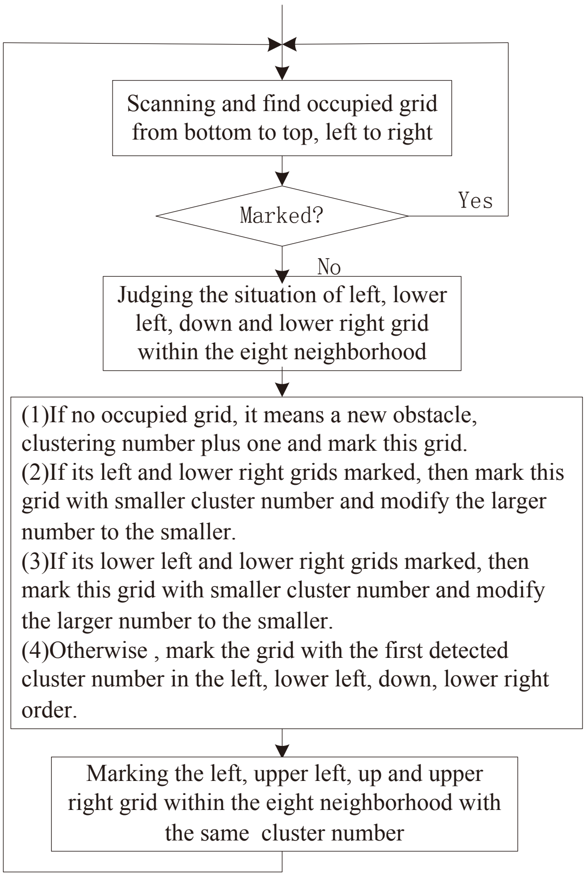

Scanning and find occupied grid from bottom to top, left to right
Marked?
Yes
No
Judging the situation of left, lower left, down and lower right grid within the eight neighborhood
(1)If no occupied grid, it means a new obstacle, clustering number plus one and mark this grid.
(2)If its left and lower right grids marked, then mark this grid with smaller cluster number and modify the larger number to the smaller.
(3)If its lower left and lower right grids marked, then mark this grid with smaller cluster number and modify the larger number to the smaller.
(4)Otherwise , mark the grid with the first detected cluster number in the left, lower left, down, lower right order.
Marking the left, upper left, up and upper right grid within the eight neighborhood with the same cluster number

Fig.6: Improved eight neighborhood region labeling algorithm

This paper proposed an improved eight neighborhood region labeling algorithm, first asks whether the grid has been marked, and then scan their neighborhood, which can effectively avoid repeating search neighborhood. Experimental results show that this method can significantly improve efficiency. Improved eight neighborhood region labeling algorithm flow chart is shown in Fig.6.

5 Experiment Results

Fig.7 is the results after different algorithms processing in the same frame. As shown in Fig.7.a, 1 is a dynamic vehicle. 2 is two dynamic pedestrians and a tricycle equipped with cargo. Since the distance is so close between pedestrians and tricycle that the algorithm will handle it together for the same obstacle, which has no effect on autonomous driving of unmanned vehicle.

5.1 Grid Map Establishment Result

In Fig.7.b, 3 are data points that scan to the ground by laser radar. According to the sensor model, see data points that more than 10cm as the barriers. Therefore, 3 will be processed for free grid in Fig.7.c.

Fig.7.c is grid map built with DSmT. The 80m×32m space is divided by 400×160 cells with a size of 0.2m×0.2m in scanning and global grid map. The miss detection rate λ_1 and false alarm rate λ_2 are all 0.1. Conflict coefficient threshold ε_1 and ε_2 are all 0.1. In Fig.7.c, the white grid is free, the green grid is occupied, the gray grid is unknown, the red grid represents $C_1 > \varepsilon_1$, that is some dynamic obstacles move into the grid, the blue grid represents $C_2 > \varepsilon_2$, that is some dynamic obstacles move out the grid.

5.2 Road Segmentation Result

In Fig.7.d, two blue lines are road edge obtained from the road edge detection algorithm, which divide the road into drivable area and undrivable area. In undrivable area, the red grid is revised as occupied, the blue grid is revised as free, and the green and gray grids remain unchanged. Which can eliminates disturbances on the dynamic obstacle detection in undrivable area.

In Fig.7.c, few blue grid of obstacle 2 can be displayed, and blue grid of obstacle 1 cannot be displayed. This is because the conflict coefficient C2 has something to do with relative velocity, the relative position, obstacle size, grid size and so on. Therefore, the C2 information cannot be stable obtained, and the moving direction of obstacles cannot be obtained through conflict coefficient.

Fortunately, the conflict coefficient C_1 can quickly reflect grid states, so the red grid should be regarded as the position of dynamic obstacles.

5.3 Expansion and Erosion Result

In Fig.7.d, there are many fracture on obstacle 1, which lead to it is difficult to cluster and track dynamic obstacles. After we implemented dilation and erosion algorithm, fracture of obstacle 1 and 2 were mended, just like Fig.7.e. This makes it easier to cluster and track dynamic obstacles.

5.4 Dynamic Obstacle Detection and Information Extraction Result

In Fig.7.f, We apply the improved eight neighborhood region labeling algorithm to cluster dynamic obstacles in drivable area, and the same clusters were wrapped together with hollow blue rectangle. The clustering number of dynamic obstacle 1 is No. 1. The distance 15.62m between dynamic obstacle 1 and unmanned vehicle was displayed below the blue box, which will move simultaneously with obstacle 1. The length of obstacle 1 is 1.60m, the width of obstacle 1 is 1.20m, which were shown in lower left.

The clustering number of dynamic obstacle 2 is No.2. The distance 23.63m between dynamic obstacle 2 and unmanned vehicle was displayed below the blue box, which will move simultaneously with obstacle 2. The length of obstacle 2 is 0.40m, the width of obstacle 2 is 0.80m, which were shown in lower right.

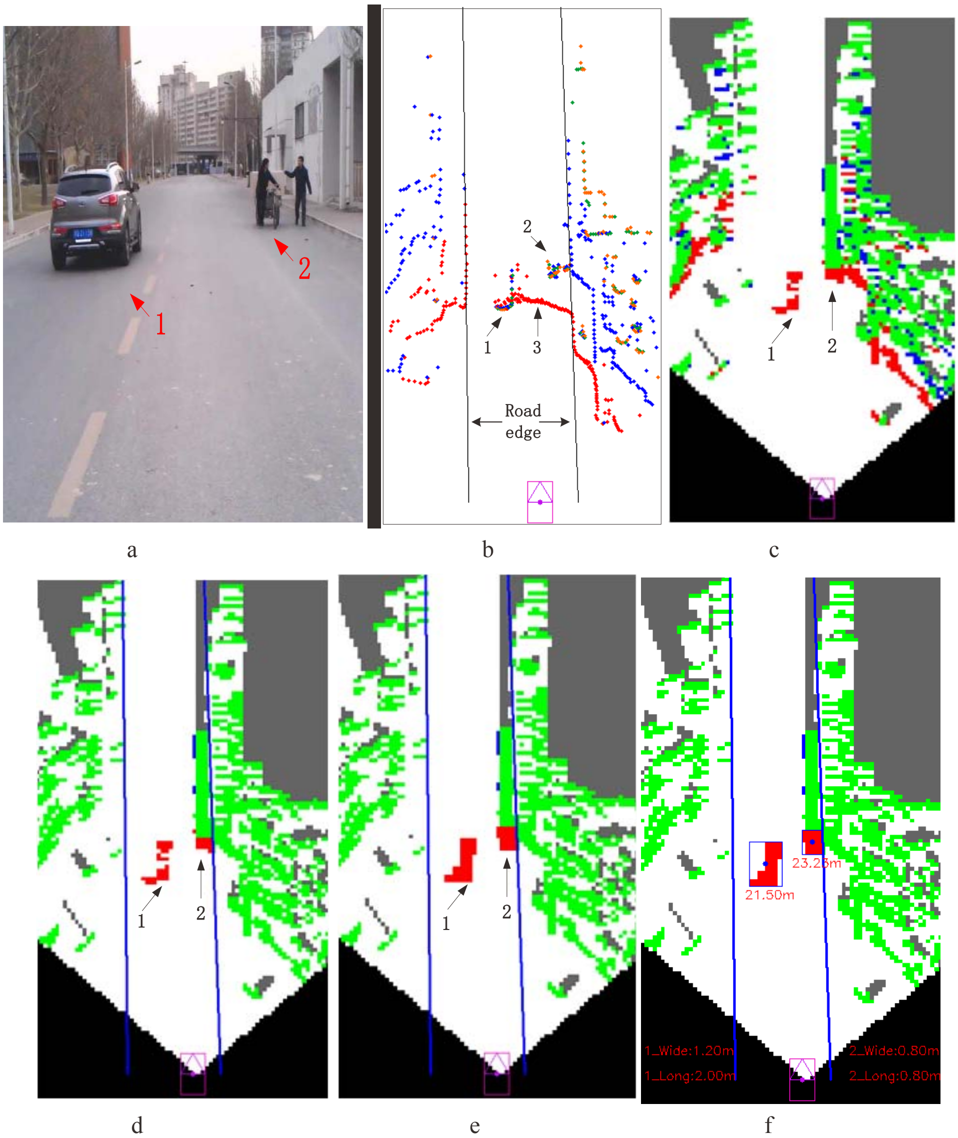


Fig.7: Dynamic obstacles detection results

6 Conclusion

In this paper we have proposed a road edge detection algorithm. Firstly the road edge data set was extracted from numerous laser radar data, and then the cluster analysis of the data set was done with the improved COBWEB algorithm based on Euclidean distance. The left and right road edge was respectively fitted a straight line with the least squares method finally. Besides, DSmT was applied to establish a grid map for the environment, and dynamic obstacles were detected by the conflict coefficient within drivable area. The cluster analysis and information extraction of dynamic obstacles were completed by the expansion algorithm, erosion algorithm and improved eight neighborhood labeling algorithm. The results show that the algorithm can stable and accurately perceive road edge and dynamic obstacles.

References

[1] S Gim, I Meo, Drivable Road Recognition by Multilayered Laser radar and Vision, Intelligent Autonomous Systems, vol. 12, pp. 43-56, 2013.

[2] X.J Yang, Detection and tracking of road based on a four-layer laser radar, M.S. thesis, Dept. Info., Zhejiang Univ., Zhejiang, China, 1993.

[3] F Yang, Z Zhu, Real-time dynamic obstacle detection and tracking using 3D Laser radar, The journal of Zhejiang University, no. 09, pp. 18-24, 2012.

[4] Y Yang, Research on an improved COBWEB algorithm, M.S. thesis, Dept. Electron. Eng., Harbin Engineering Univ., Heilongjiang, China, 2010.

[5] Q Du, B.L Hang, Application of least squares in multisensor measure specifying, The journal of sensing technology, no. 02, pp. 36-38, 2005.

[6] H.Y Cao, H.X Sun, Building grid map with improved Dempster-Shafer evidence theory, The journal of Jilin University, no. 04, pp. 239-243, 2011.

[7] J Moras, V Cherfaoui, Credibilist Occupancy Grids for Vehicle Perception in Dynamic Environments, 2011 IEEE International Conference on Robotics and Automation, Shanghai, 2011, pp. 84-89.

[8] J Hou, M Zhuang, A adaptive integration algorithms with DST and DSmT, Microelectronics and Computer, vol.23, no.10, pp. 150-152, 2006.

[9] Y.Z Zhang, H.S Zhao, New method for component-labeling in binary image, Application Research of Computers, no.11, pp. 342-344, 2010.

[10] Z.Z Luo, Y.W Zhou, Optimizing the algorithm of labeling connected components based on region-growth, The journal of Minjiang University, no.02, pp. 47-50, 2011.