Pedestrian Detection and Tracking Based on 2D Lidar

Jiannan Chen, Ping Ye
Department of Automation
Beijing University of Posts and Telecommunications
Beijing, China

Zhipeng Sun
Department of Electrical Engineering
Northeast Electric Power University
Jilin, China

Abstract—**In order to make the service robot carry out dynamic path planning according to the movement state of pedestrians and improve the ability of collaboration with humans, this paper proposes a pedestrian detection and tracking algorithm based on 2D lidar. Pedestrian detection based on 2D lidar is mainly done by detecting the legs of pedestrians. In the laser data preprocessing stage, in order to reduce the influence of lidar point cloud noise, Gaussian filtering is used to filter the original point cloud data. In order to reduce the influence of the environment itself on the detection accuracy, an algorithm for projecting the laser data with the static map is proposed to remove the static background data. In the clustering segmentation stage, this paper uses Euclidean clustering to segment the laser point cloud clustering. In the feature extraction stage, in order to improve the detection accuracy of pedestrians' legs, the number of statistical features is increased on the basis of extracting geometric features; In order to improve the generalization ability of the model, and to solve the problem of scale inconsistency of pedestrian leg feature description, the extracted pedestrian features are normalized. This article uses SVM to build a human leg classifier. In the pedestrian tracking phase, this paper proposes a multi-person tracking algorithm based on Kalman filtering. The experimental results show that the proposed scheme can obtain better pedestrian detection and tracking effects, and the real-time performance is higher.**

Keywords-component; 2D lidar; pedestrian detection; SVM; tracking; kalman; laser background removal

I. Introduction

The technical problems that need to be solved in the normal operation of service robots are location, SLAM and navigation [2]. Navigation is a basic capability of service robots. Most of the traditional robots work in a static environment. In this case, you only need to do the corresponding path planning in the pre-learned environment map. You don't need to deal with dynamic obstacles, and you rarely need to interact with people. The main task of the service robot is to deal with people. Whether it is to follow people or to navigate in pedestrian environment, it is necessary to identify the pedestrians in the environment. To detect and track pedestrians, robot first need to classify the environment and distinguish between pedestrians and non-pedestrians. Then it is necessary to accurately discriminate the position of the pedestrian, and at the same time, it is necessary to distinguish different pedestrians and establish a mathematical model to accurately track. Tracking pedestrians is an estimate of the state of the pedestrian. Adding the movement state of the pedestrian to the decision of the path planning can make the navigation behavior of the service robot more friendly and more natural.

In the field of pure pedestrian detection, pedestrian detection based on deep neural networks has been able to achieve high detection accuracy[13][14][15]. However, this algorithm is very demanding on the processor, which will greatly increase the cost. In the field of unmanned driving, multi-line lidars are often standard sensors. Since multi-line lidars can acquire more information than single-line lidars, the use of multi-line lidars to detect pedestrians can often achieve higher accuracy than single-line lidar detection [3]. However, due to the high price of multi-line lidar, it is rarely used in the field of mobile robots. For the single-line lidar, which is standard in service robots, based on the principle of reducing the research and development cost of service robots, this paper proposes a pedestrian detection and tracking solution based on 2D lidar.

Pedestrian detection based on 2D lidar is mainly done by detecting the legs of pedestrians [4]. Since the point cloud data of the lidar itself contains a certain amount of noise, extracting features in the point cloud data containing noise tends to deviate from the real result. In order to solve the laser point cloud noise problem, the original laser point cloud data is filtered by Gaussian filtering [5]. Pedestrian legs are mainly arc-like features. Because 2D lidar itself acquires less information, when there are more static arc-shaped objects in the environment, the detection accuracy and tracking effect will be greatly reduced. In order to solve the influence of static background arc objects on detection accuracy, this paper proposes an algorithm for matching laser point cloud and static map projection, which can effectively remove static background point cloud data. In the process of detecting pedestrians by lidar, the legs of the pedestrians tend to merge. At this time, the shape of the point cloud that is fed back is not a circular arc shape, and the shape of the pedestrian's legs can not be well described by geometric features alone. In order to solve this problem, the proportion of statistical features is improved in the feature vector. In the human leg feature vector, there are different scale descriptions, such as curvature, point cloud region width and variance are not on a scale. The generalized ability is often poor with models trained by different scale features. In order to solve the problem of feature scale inconsistency in the feature vector, the normalized feature vector is processed [6]. SVM is a two-class model [7]. This paper uses SVM to construct a human leg classifier. Kalman filtering is an algorithm that uses the linear system state equation to estimate the state of the system through the input and output of the system [8]. In this paper, the pedestrian motion in the adjacent state is approximated as a uniform motion. In order to solve the problem of mismatching the same ID pedestrian in the multi-walker tracking process, based on the distance judgment, the judgment of pedestrian speed and motion direction is added.

II. RELATED WORK

Currently, the main methods of pedestrian detection are vision-based pedestrian detection and lidar-based pedestrian detection. Pedestrian detection models based on deep neural networks include SSD, YOLOv3 [9], etc., which generally have higher detection accuracy than traditional machine learning models, but cannot be run on mobile robot controllers in real time. The working environment of the lidar sensor is not affected by light, and the measurement accuracy and speed are high, which is suitable for the field of mobile robots. Since the mobile robot's lidar sensor is installed close to the ground, the lidar sensor detects pedestrians mainly relying on the detection of pedestrians' legs. In recent years, multi-line lidar has been widely used due to the rise of the unmanned field. Multi-line lidars have more information than single-line lidars, so using multi-line lidar often results in higher accuracy. Multi-line lidar detection of pedestrians, one method is to cluster the point cloud data, and then extract features, using traditional machine learning classification network for classification; another method is to use deep neural network to directly classify point cloud data. Both solutions can achieve good detection results, but multi-line lidar is not a standard sensor for mobile robots, and the use of multi-line lidar often requires a higher-end processor, which will bring high cost to mobile robots.

Based on state estimation tracking, the commonly used algorithms are Particle filtering and Kalman filtering. In highly nonlinear systems, the accuracy and stability of Particle filter algorithms have certain advantages [1]. However, as the particle number of the Particle filter algorithm increases, the calculation time will increase in an approximate series, and the real-time performance of the system will be affected. The Kalman filter algorithm performs well in the case where the mathematical model of the system is relatively simple and the predicted value obeys the Gaussian distribution. Applied to multi-walker tracking based on lidar, most Kalman filters are based on the detection distance of the previous moment and the current moment. In the case of detecting a short-term failure or a short-term occlusion, mis-tracking occurs.

III. SYSTEM OVERVIEW

The entire system consists of four modules, including cluster segmentation, feature extraction, human leg classifier and multi-person Kalman filter tracker. The original laser point cloud passes Gaussian filtering to reduce the influence of lidar point cloud noise on feature extraction. After the static laser point cloud background is removed, there is a point cloud that cannot match the static map, which is the new pedestrian point cloud in the current scene. The remaining point clouds are clustered and the objects in the environment are separately segmented. Feature vectors are obtained by extracting geometric features and statistical features from each segment of the laser point cloud. The feature vector corresponding to the laser point cloud region is classified by the trained human leg classifier to realize the human leg and the non-human leg. The classified human leg position data is used as an input to the multi-person Kalman filter tracker. Finally, the multi-person Kalman filter tracker outputs the position, velocity and direction of movement of the pedestrian in the current scene.

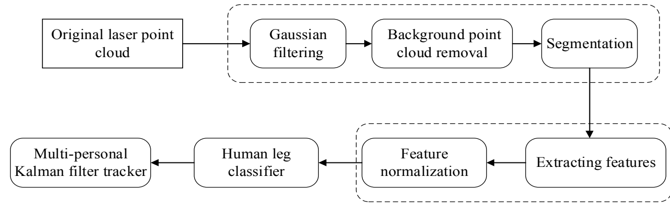

Figure 1. System block diagram

IV. PEDESTRIAN DETECTION

A. Cluster Segmentation

1) Gaussian filtering

There are many sources of noise for lidar measurement, such as the frequency interference between the ambient light source and the lidar light pulse; the specular reflection caused by the laser pulse hitting the smooth material; the error of the lidar itself. Due to the above factors, the accuracy of the lidar measurement will be degraded. If the collected data is directly processed, it will bring a large error, which will affect the detection accuracy of the subsequent algorithm. Therefore, the original point cloud collected by the lidar must be filtered to reduce the influence of noise.

In this paper, Gaussian filtering is used to preprocess the data. Gaussian filtering is widely used in the field of signal processing as a linear smoothing filter. Each current data point is weighted and averaged by itself and its neighboring data points. The weighted average of [0.25, 0.5, 0.25] is used to weight the average of three adjacent points. Let P be the point before filtering, P' be the filtered point, $\mathrm{P}-1$, $\mathrm{P}-2$ be its neighboring points, and use the index boundary point instead for the point beyond the index range.

$$P' = \begin{bmatrix} \mathrm{P}-1 \\ P \\ P+1 \end{bmatrix} [0.25 \quad 0.5 \quad 0.25] \tag{1}$$

2) Static laser point cloud background removal

The pedestrian detection and tracking in this paper runs in the environment where the mobile robot is known to locate, and can subscribe to the static map of the current scene. The coordinate system of the static map is /map, the coordinate system of the mobile robot is /base_link, and the coordinate system of the lidar is /laser_link. The coordinate conversion relationship ${}^{b}_{l}T$ of /laser_link to /base_link is static coordinate conversion, which is determined by the installation position of the lidar. The coordinate conversion relationship ${}^{m}_{b}T$ of /map to /base_link is obtained by the positioning system of the mobile robot. The coordinates of the laser point cloud in the /laser_link coordinate system ${}^{l}_{p}P_i$ can be expressed as $({}^{l}_{p}x_i, {}^{l}_{p}y_i, 0)^T$, the coordinates of the laser point cloud in the /map coordinate system ${}^{m}_{p}P$ is:

$${}^{m}_{p}P_i = {}^{m}_{b}T\,{}^{b}_{l}T \begin{bmatrix} {}^{l}_{p}P \\ 1 \end{bmatrix} \tag{2}$$

The value of the static raster map in ROS [11] is represented by -1, 0-100, where -1 means the grid is unknown, 0 means the grid is not occupied, and 100 means the grid is completely occupied. The strategy adopted in this paper is to search for static grid maps within the range of laser point cloud world coordinates ${}^{m}_{p}P_i$ threshold P_thres. If a grid with a cost of 100 is searched for a radius of P_thres centered on the world coordinate ${}^{m}_{p}P$, the current laser spot is considered to match the static map as a static laser point cloud. The accuracy of the grid map is 0.05m, and the positioning of the mobile robot will also have a certain deviation. In this paper, the search threshold P_thres is 0.05m.

3) Euclidean clustering

The main principle of Euclidean clustering is to classify point cloud data within a certain distance threshold range into one category until there is no other point cloud data in the current clustering threshold range [12]. The laser data clustering segmentation is based on the Euclidean distance of the two-dimensional coordinates of the adjacent laser points in the world coordinate system. If D(P_i,P_{i+1})is greater than the set threshold D_{th}, the current laser data should be considered to be segmented:

$$\mathrm{D}(P_{i+1}, P_i) = \sqrt{(X_{i+1} - X_i)^2 + (Y_{i+1} - Y_i)^2} \qquad (3)$$

Although the principle of Euclidean segmentation is relatively simple, testing in actual scenes can achieve good results.

B. Feature Extraction

1) Define features

After clustering and segmenting the laser data, this paper defines a series of features for the laser data segment to classify the human legs. The number of laser points in the segmentation area; the average angle difference of the segmentation area, that is, the average value of the angle difference between the lines of the adjacent two laser point cloud data in the segmentation region; the radius of the arc of the segmentation region, by fitting the point cloud data of the segmentation region A circle is used, and the radius of the circle is used instead of the radius of the arc; the length of the boundary of the segment is divided, and the circle length is calculated by the circle fitted by the point cloud data and the center angle corresponding to the boundary of the region; the width of the segment, that is, the cluster region The distance between the endpoint cloud coordinates; the average curvature of the segmentation region; the linearity of the segmentation region, the fitting of a line with the laser point cloud coordinates in the region, and the sum of the distances from the laser point cloud coordinates to the line; the mean value of the segmentation region coordinates Difference; the standard deviation of the median value of the segmentation region; the variance of the segmentation region coordinates; the kurtosis of the segmentation region; the skewness of the segmentation region [10].

Among the 12 features defined herein, features 1-7 describe the geometric features of the laser data segment, describing the geometry of the human leg. Features 8-12 are statistical features of the laser data segment, where features 8-10 describe the degree of dispersion of the laser data, and features 11-12 describe the morphology of the laser data, i.e., the degree of soarness and degree of symmetry.

2) Feature normalization

In the objective function of the machine learning algorithm, the basis of the objective function in many learning algorithms is to assume that all features are zero mean and have variances on the same order. If the variance of a feature is orders of magnitude larger than other features, it will dominate the learning algorithm, causing the learner to learn from other features as we would expect. We normalize the eigenvectors using Gaussian normalization:

$$F_i' = \frac{F_i - \bar{X}}{\frac{1}{n}\sum_{i=1}^{n}(F_i - \bar{X})^2} \qquad (4)$$

Among them, F_i is one of the features of the feature vector, $\bar{X}$ is the mean of the feature vector, n is the number of feature vectors, and F_i' is the normalized feature.

C. Classification

Support vector machines (SVM) are a two-class classification model. The basic model is defined as the linear classifier with the largest interval in the feature space. The learning strategy is to maximize the interval. The maximum interval makes it different from the perceptron. The support vector machine also includes the nuclear technique, which makes it a substantial Nonlinear classifier. The learning strategy of support vector machine is to maximize the interval, which can be formalized as a problem of solving convex quadratic programming, and is also equivalent to the minimization of regularized hinge loss function. The learning algorithm of support vector machine is the optimization algorithm for solving convex quadratic programming.

For the problem of nonlinear separability, the data can be transformed into a linear separable problem by selecting appropriate kernel functions to map the data into a higher-dimensional feature space. Therefore, the maximal equation for solving the optimal hyperplane for the nonlinear separable problem is:

$$\min_{\alpha} W(\alpha) = \sum_{i=1}^{l} \alpha_i - \frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l} \alpha_i\, \alpha_j y_i y_j K\langle x_i \bullet x_j \rangle \qquad (5)$$

Optimal classification discriminant function:

$$\mathrm{f(x)} = \mathrm{sgn}\left(\sum \alpha_i y_i K(x \bullet x_i) + b\right) \qquad (6)$$

Where $K(x \bullet x_i)$ is a kernel function. Different types of SVM can be constructed by selecting different kernel functions. Considering that the data has no prior knowledge and simplifies the calculation process, the kernel function chosen in this paper is the Gauss kernel function:

$$\mathrm{K}(||x_i - x_c||) = \exp\left(\frac{-||x_i - x_c||^2}{2\sigma^2}\right) \qquad (7)$$

Where x_c is the center of the kernel function; σ is the width parameter of the function, and the radial range of the function is controlled.

V. PEDESTRIAN TRACKING

A. Kalman Filter Mathematical Modeling

Kalman filtering is an algorithm that uses the linear system state equation to estimate the state of the system through the input and output of the system. Since the observed data includes the effects of noise and interference in the system, the optimal estimate can also be considered as a filtering process. The use of Kalman filtering needs to satisfy three conditions: 1) the dynamic system is a linear system; 2) the observation is a linear combination of states; 3) the noise obeys a normal distribution. In the implementation of the pedestrian tracking algorithm in this paper, it is assumed that the pedestrian is a constant speed model in a short time, and the laser measurement noise obeys a normal distribution.

The state vector of the pedestrian can be expressed as $X_t = (P_{xt} \quad P_{yt} \quad V_{xt} \quad V_{yt})^T$. The status of the current moment can be expressed as:

$$P_{xt} = P_{xt-1} + V_{xt-1}\Delta t \tag{8}$$

$$P_{yt} = P_{yt-1} + V_{yt-1}\Delta t \tag{9}$$

$$V_{xt} = V_{xt-1} \tag{10}$$

$$V_{yt} = V_{yt-1} \tag{11}$$

After writing in matrix form:

$$\begin{bmatrix} P_{xt} \\ P_{yt} \\ V_{xt} \\ V_{yt} \end{bmatrix} = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} P_{xt-1} \\ P_{yt-1} \\ V_{xt-1} \\ V_{yt-1} \end{bmatrix} \tag{12}$$

Further extract the state transformation matrix:

$$A = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{13}$$

The formula can be simplified to:

$$\bar{X}_t = \mathrm{A}X_{t-1} \tag{14}$$

Where A is the state transition matrix.

The transfer formula of the noise covariance matrix is: $\bar{P}_t = \mathrm{A}P_{t-1}A^T + Q$. Q is a covariance matrix, which represents the noise caused by the prediction model itself.

The observation vector is $Z_t = (P_{xt} \quad P_{yt})^T$, and the transformation relationship from the state vector of the pedestrian itself to the observed state vector is denoted as H. Since the transformation is a linear relationship, H can be represented by a matrix, that is, an observation matrix. Since the observations also contain noise, v is added to indicate the observed noise, and the covariance matrix of the noise is represented by R.

$$Z_t = HX_t + \mathrm{v} \tag{15}$$

$$\mathrm{H} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \tag{16}$$

$Z_t - H\bar{X}_t$ represents the residual between the actual observation and the predicted value, and K_t represents the Kalman coefficient, which is used to correct the $\bar{X}_t$ value.

$$X_t = \bar{X}_t + K_t(Z_t - H\bar{X}_t) \tag{17}$$

$$K_t = \bar{P}_t H^T (H\bar{P}_t H^T + R)^{-1} \tag{18}$$

Update the noise distribution of the best estimate, which is left for the next iteration. The uncertainty of the state of this step is decreasing, and in the iteration of the next round, the uncertainty increases again due to the introduction of the transmitted noise. Kalman filtering is looking for a balance in this ever-changing noise.

$$P_t = (\mathrm{I} - K_t H)\bar{P}_t \tag{19}$$

B. Multi-Pedestrian Tracker Design

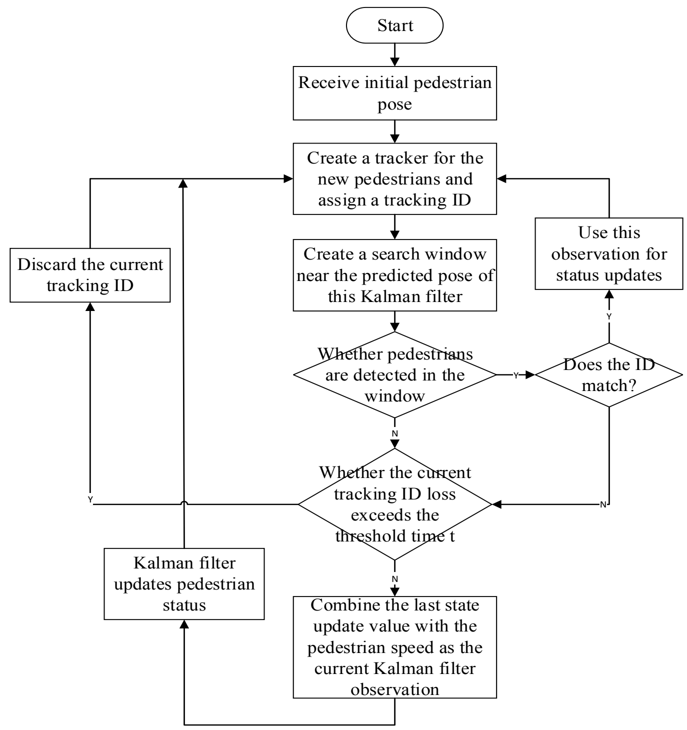

Figure 2. Multi-target tracking flowchart

The problems that multi-target (pedestrian) tracking needs to solve relative to single-target tracking are as follows: 1) Target matching: By comprehensively matching the speed, direction of motion and distance of the pedestrian state of the preceding and succeeding frames, the pedestrian matching relationship of the highest score before and after is taken as The same ID pedestrian. 2) A new target appears: by matching the pedestrian detected by the current frame with the pedestrian state predicted by the Kalman filter, if not correctly matched, the new target is considered to be added. 3) Target occlusion: If a pedestrian of an ID is not detected within a certain time range, but does not exceed the threshold time t, then the pedestrian is considered to be occluded. 4) Target disappears: If the pedestrian of an ID exceeds the threshold time t is not detected, the current target is considered to have disappeared.

VI. Experimental Result

We test the performance of the algorithm through the public pedestrian detection data set and our own mobile robot platform as shown in Fig. 3. The HOKUYO 2D laser lidar is built on the mobile robot platform. The Lidar measurement accuracy is ±4cm, the measurement range is 270°, the resolution is 0.25°, the measurement frequency is 40 Hz, and the ranging range is 10 m.

Figure 3. Mobile robot

Figure 4. Lidar

A. Gaussian Filtering

We used a mobile robot platform to collect a set of laser data on the wall. Fig. 5 shows the laser point cloud before filtering, and Figure 6 shows the filtered laser point cloud. After filtering, the overall smoothness of the laser point cloud is obviously improved, and the original noise of the laser point cloud is weakened.

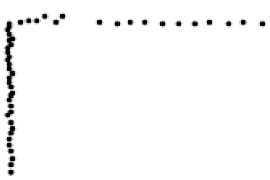

Figure 5. Cloud before filtering

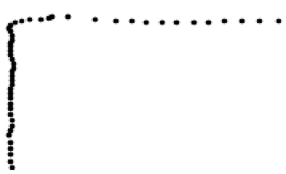

Figure 6. Filtered point cloud

B. Pedestrian Test Data Set Test

The training data set we used had 150,515 positive samples and 975,473 negative samples. The test data set has 95,298 positive samples and 336,265 negative samples. In Table 1, T1 represents the features currently used by the existing method. T2 represents the effect of adding statistical features on the basis of T1. T3 represents the effect of adding feature normalization on the basis of T2. T4 represents the effect of adding laser point cloud filtering on the basis of T3. The performance evaluation indicators used in this article are: precision (P), recall (R) and F1 measure.

$$\mathrm{P} = \mathrm{TP}/(\mathrm{TP} + \mathrm{FP}),\ \mathrm{P} \in [0,1] \tag{20}$$

$$\mathrm{R} = \mathrm{TP}/(\mathrm{TP} + \mathrm{FN}),\ \mathrm{R} \in [0,1] \tag{21}$$

$$\mathrm{F1} = 2 * \mathrm{P} * \mathrm{R}/(\mathrm{P} + \mathrm{R}),\ \mathrm{F1} \in [0,1] \tag{22}$$

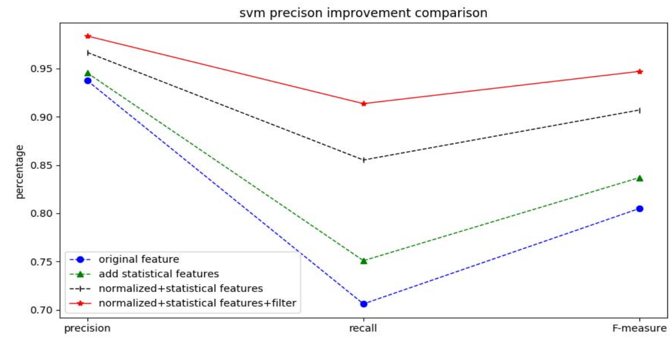

Figure 7. Accuracy comparison

TABLE I. COMPARISON OF METHOD ACCURACY PROPOSED IN THIS PAPER

Test items	Precision	Recall	F1-measure
T1	0.937	0.706	0.805
T2	0.945	0.751	0.837
T3	0.967	0.855	0.907
T4	0.984	0.914	0.947

C. Removing the Static Laser Point Cloud

Fig. 8 shows the experimental environment in which there are many impurities and it is prone to false detection. Fig. 9 shows the pedestrian detection effect without removing the static laser background, where the green sphere represents that the laser point cloud is detected as a pedestrian. In fact, only the places enclosed by the boxes in the picture are real pedestrians, and other places are false detections. Fig. 10 shows the pedestrian detection effect after removing the background of the static laser data. The white laser point cloud represents the static laser point cloud matched with the map, the red laser point cloud is the remaining dynamic laser point cloud, and the green sphere indicates the detected pedestrian. It can be seen that the pedestrian detection after removing the static laser point cloud can significantly improve the pedestrian detection accuracy.

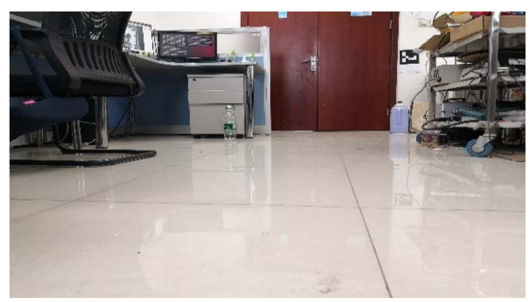

Figure 8. Experimental environment

Figure 9. Detection effect before removing background laser point cloud data

Figure 10. Detection effect after removing background laser point cloud data

D. Pedestrian Tracking

Fig. 11 is a relatively messy indoor pedestrian tracking in the background. Fig. 12 shows the pedestrian tracking of the corridor environment test. It can be seen that the method proposed in this paper can effectively estimate the position, speed and direction of movement of pedestrians.

Figure 11. Indoor messy environment pedestrian tracking effect

Figure 12. Empty channel pedestrian tracking effect

All experiments were performed on an industrial computer equipped with a 2.6 GHz i5 CPU on a mobile robot. The system can run at frequencies up to 40 Hz, meeting the high real-time requirements on low-end processors.

VII. CONCLUSIONS

This paper proposes a method to detect pedestrians in a messy background using 2D lidar to achieve higher detection accuracy. At the same time, a multi-person tracking algorithm based on Kalman filter is proposed to effectively reduce pedestrian mismatch. The original laser point cloud is not filtered by Gaussian filtering, which reduces the original noise of the laser point cloud. The proposed matching algorithm with static raster map can effectively reduce the impact of non-pedestrian static environment on pedestrian detection accuracy. On the basis of geometric features, increasing the number of statistical features can effectively improve the detection effect of pedestrians' legs. The normalization of the feature vector effectively improves the generalization ability of the model and the detection accuracy. In the pedestrian tracking phase, this paper uses the pedestrian position, speed and direction of motion to make a comprehensive match to associate the same ID pedestrians, effectively avoiding pedestrians' false tracking.

REFERENCES

[1] Gaddigoudar, K. Prateek, Balihalli. "Pedestrian detection and tracking using particle filtering." International Conference on Computing IEEE, 2017.

[2] Vosselman, George, M. Coenen, and F. Rottensteiner. "Contextual segment-based classification of airborne laser scanner data." ISPRS Journal of Photogrammetry and Remote Sensing 128(2017):354-371.

[3] Tang, Hsueh Ling, Chien. "Multi-cue pedestrian detection from 3D point cloud data." IEEE International Conference on Multimedia & Expo IEEE, 2017.

[4] D. Li, L. Li, Y. Li. "A Multi-type Features Method for Leg Detection in 2D Laser Range Data." IEEE Sensors JournalPP.99(2017):1-1.

[5] W. Guoqing, G. Zhongxing, Z. Yonggang. "Adaptive Maximum Correntropy Gaussian Filter Based on Variational Bayes." Sensors 18.6(2018):1960-.

[6] J. Fan, F. Jing, Z. Fang. "Automatic recognition system of welding seam type based on SVM method." International Journal of Advanced Manufacturing Technology 92.1-4(2017):1-11.

[7] R. A. Kharjul, V. K. Tungar, Y. P. Kulkarni. "Real-time pedestrian detection using SVM and AdaBoost." 2015 International Conference on Energy Systems and Applications IEEE, 2015.

[8] Chong, Penny , and Y. H. Tay . "A novel pedestrian detection and tracking with boosted HOG classifiers and Kalman filter." 2016 IEEE Student Conference on Research and Development (SCOReD) IEEE, 2016.

[9] Redmon, Joseph, and A. Farhadi. "YOLOv3: An Incremental Improvement." (2018).

[10] Navarro P J, Carlos Fernández, Raúl Borraz. "A Machine Learning Approach to Pedestrian Detection for Autonomous Vehicles Using High-Definition 3D Range Data." Sensors 17.1(2016):18.

[11] Mishra, Ruchik, and A. Javed. "ROS based service robot platform." 2018 4th International Conference on Control, Automation and Robotics (ICCAR) IEEE, 2018.

[12] Bouhmala, Noureddine. "How Good is the Euclidean Distance Metric for the Clustering Problem." 2016 5th IIAI International Congress on Advanced Applied Informatics (IIAI-AAI) IEEE Computer Society, 2016.

[13] J. Ker, L. P. Wang, J. Rao, and T. Lim. "Deep Learning Applications in Medical Image Analysis." IEEE Access, vol.6, pp. 9375-9389, 2017.

[14] X. Bai, Y. Fang, W. Lin, L.P. Wang, and B.-F. Ju. "Saliency-based Defect Detection in Industrial Images by Using Phase Spectrum." IEEE Trans. Industrial Informatics, vol.10, no.4, pp. 2135-2145, 2014.

[15] Lining Zhang, L. P. Wang and Weisi Lin. “Geometric Optimum Experimental Design for Collaborative Image Retrieval.” IEEE Trans. Circuits and Systems for Video Technology, vol.24, pp.346-359, 2014.