Point cloud target detection and tracking algorithm based on K-means and Kalman

Liangyu Fang\textsuperscript{a}, Qian Luo\textsuperscript{b} Yue Sun

Information and Communication Institute, Beijing Information Science Technology University, Beijing 100101, China
\textsuperscript{a}fangliangyu@beyondcent.com, \textsuperscript{b}luoqian@bistu.edu.cn

Abstract. This paper is mainly about point cloud image and data processing, virtualization and visualization. An improved K-means iterative clustering algorithm is proposed. In addition, cloth simulation filtering algorithm is used to extract road surface information. On this basis, bilateral filtering and statistic filtering are used to smooth image denoising. Kalman filter is used to track and predict point cloud targets. Using the variance of innovation sequence, the noise variance and observation noise variance can be estimated and corrected gradually in the process of system calculation. The error is relatively small, and the accuracy and reliability are improved. For disordered, occluded, sparsity and noisy, complex terrain image, how to improve the denoising effect, the accuracy and effectiveness of target extraction and tracking is the focus of this paper.

Keywords: Target detection and tracking, K-means, filtering denoising, Kalman filter, anti-crawling.

1. Introduction

Popular sensors such as rgb-d depth camera, lidar can obtain point cloud. In view of the uncertainty and complexity of the target, this paper proposes a target detection and tracking algorithm as well as a few anti-crawlers [8] strategies.

2. Point cloud target detection based on K-means and filtering denoising algorithm

Firstly, the cloth simulation filtering CSF algorithm [1] separates the ground points from the original point cloud [1,2]. By analyzing the interaction between the particle nodes and the corresponding points of the adjacent point cloud, the program can separate the ground and non-ground points. After that, the denoising filtering method is used to process several frames of point cloud image to achieve the purpose of image smoothing and noise reduction [2]. Statistical filtering is used here: the average distance from each point to all adjacent points is calculated. If the Gaussian distribution is obtained, the points whose average distance is outside the standard range are outliers, which are removed. Bilateral filtering is used to save the pixel values near the edge. It not only effectively reduces the noise, but also maintains the geometric feature information to avoid being excessively smooth and edge preserving denoising.
Under the same light conditions, the reflectance of bright surface is high, and that of dark objects such as asphalt and wet soil is low. In this paper, Euclidean distance is used to cluster the echo intensity information. Distance between $\mathbf{x} = (x_1, x_2, \ldots, x_p)^T$ and $\mathbf{y} = (y_1, y_2, \ldots, y_p)^T$:

$$D(x, y) = \sqrt{(x_1 - y_1)^2 + \ldots + (x_p - y_p)^2}$$

(1)

Euclidean distance calculation requires that all dimension data are in the same dimension level; the distribution of each component may be different, ignoring the influence of correlation; noise has great influence on Euclidean distance calculation, and the anti-interference ability of the distance is poor. Therefore, this paper improves the algorithm. For the Multidimensional variables whose covariance matrix is $S$ and mean value is $\mu = (\mu_1, \mu_2, \ldots, \mu_p)^T$, the Mahalanobis distance is calculated as

$$D_M(x) = \sqrt{(x - \mu)^T S^{-1} (x - \mu)}$$

(2)

If $S$ is a unit matrix, then it is Euclidean distance. Mahalanobis distance avoids the interference of correlation between dimensions and variables. The distance evaluation function (DEF) is introduced to select the most appropriate clustering method to extract the road surface information. The Mahalanobis distance of each cluster center is calculated, and it is classified into the nearest centroid class. The steps are as follows:

1) The number of point cloud data is $m \{x_1, x_2, \ldots, x_m\}$. Output $K$ cluster centers $\{C_1, C_2, \ldots, C_k\}$, $K$ cluster centers are formed.

2) The Mahalanobis distance of each cluster center is calculated, and it is classified into the nearest centroid class.

3) Recalculate the centroid of each class.

The calculation formula is

$$u_j = \frac{1}{|C_j|} \sum_{x \in C_j} x \quad (j = 1, 2, \ldots, k)$$

(3)

Iteration 2, 3 steps until the new centroid and the original centroid equal or less than the specified threshold, the end of the algorithm.

According to the clustering algorithm, the clustering results should meet the requirements of the highest similarity and the smallest difference within the type, the lowest similarity and the largest difference between the types, and any element after clustering is closest to its class center. The distance evaluation function (DEF) is constructed to find the optimal clustering. Define the distance within the class

$$D_{in} = \sum_{j=1}^{k} \sum_{x \in C_j} |x - u_j|$$

(4)

The distance between clusters is defined as the sum of the distances between all cluster centers and all element centers

$$D_{out} = \sum_{j=1}^{k} |u_j - u|$$

(5)

Where $x$ is the element belonging to the $j$-th cluster center, and $u$ is the mean value of all elements.

In order to achieve the optimal clustering, the minimum criterion of DEF is introduced to iterate for many times. After clustering, the point clouds with different intensity information types are obtained, which type of point cloud can be determined as the target vehicle point cloud according to the semantic information of the region, and other types of point clouds are exposed soil roads or other ground objects. Specified DEF:

$$F(k) = |D_{in} - D_{out}|$$

(6)
After denoising and filtering the selected point cloud data, 43092 ground point clouds are formed. K-means and improved k-means algorithm are used to cluster point cloud intensity information respectively, and two corresponding clustering results are formed, as shown in Table 1

| types | K-means | Improved K-means |
|-------|---------|------------------|
| 2     | 6.02    | 5.28             |
| 3     | 5.38*   | 3.76*            |
| 4     | 7.43    | 6.27             |
| 5     | 10.29   | 9.01             |
| 6     | 12.56   | 11.22            |

The results show that the distance of DEF is the smallest and the clustering effect is the best. *Represents the optimal clustering. Compared with the f(k) data in the table, f(k) value of the improved k-means clustering algorithm is smaller than that of traditional K-means clustering algorithm, which means that the clustering effect obtained by Mahalanobis distance is dominant.

3. Point cloud target detection based on Kalman filtering algorithm

The state of the original system is represented by $x_{k-1}$. The previous state quantity acts on the current state, and the known control information $U_k$ and certain noise $w_k$ are added to produce a new state. The state equation of the system is:

$$x_k = Ax_{k-1} + BU_k + w_k$$  \hspace{1cm} (7)

Among them, $A_k$ is the state transition model (some data are called system model), $B$ is the control input model, and $w_k$ is the process noise, $w_k \sim N(0,Q_k)$

At the same time, the system outputs current state $x_k$ through a linear operator $Z_k$ disturbed by noise to form the observation equation of the system (some paper call output equation)

$$Z_k = Hx_k + v_k$$ \hspace{1cm} (8)

$H$ is Observation model, $v_k$ is Observation noise, $v_k \sim N(0,R_k)$

The target motion state is described by the following formula

$$X_k = A^\star X_{k-1} + w = \begin{bmatrix}
  x_k \\
  y_k \\
  0 0 1 0^T x_{k-1} \\
  0 0 0 1^T y_{k-1}
\end{bmatrix} + w$$ \hspace{1cm} (9)

$$Z_k = H^* X_k + v = \begin{bmatrix}
  1 & 0 & t & 0 \\
  0 & 1 & 0 & t \\
  0 & 0 & 1 & 0 \\
  0 & 0 & 0 & 1
\end{bmatrix} X_k + v$$ \hspace{1cm} (10)

$x_k$ represents the state matrix of the target center position, $A$ represents the state transition matrix of the target motion, $x_k$, $y_k$ represents the coordinate values of the target center in X and Y axes, $x_{k-1}$, $y_{k-1}$ represents the velocity value of the target center, $w$ represents the process noise, $Z_k$ represents the measurement position of the target center, and $H$ represents the measurement position of the target center. $v$ represents measurement noise.

Kalman [5-7] in the prediction stage, the state prediction equation is as follows:

$$x_{k|k-1} = Ax_{k-1|k-1}$$  \hspace{1cm} (11)
$x_{k|k-1}$ is the prediction of the current state by using the previous state, and $x_{k-1|k-1}$ is the optimal result of the previous state.

Estimate $P$ in $x_{k|k-1}$ estimation state, prediction equation of covariance matrix $P$ is

$$P_{k|k-1} = AP_{k-1}A^T + Q_k$$ (12)

The system state update equation is as follows,

$$x_k = x_{k|k-1} + K_k(Z_k - Hx_{k|k-1})$$ (13)

$K_k$ is the Kalman gain of the system,

$$K_k = P_{k|k-1}H^T(HP_{k|k-1}H^T + R)^{-1}$$ (14)

Kalman gain $K$ is used to correct the prediction error. The smaller the gain is, the more reliable the estimation result is. Because Kalman filter predicts from the recursive formula, it does not need a lot of historical data and reduces calculation.

$R$ is the covariance of the measurement noise $v$. The optimal estimation of the current state has been obtained from the above calculation, but Kalman filter [5-7] is a process of recursive operation until the error is less than a certain threshold. The covariance update equation of the current state is as follows

$$P_{k|k} = (I - K_kH)P_{k|k-1}$$ (15)

The Kalman prediction of the linear model is given above, but the actual system has different degrees of nonlinearity, such as square, triangle, square root relationship, etc. the nonlinear function expression can be expanded according to Taylor series formula, omitting the high-order term, and an approximate linear model can be obtained, and then the Kalman filter recursive processing is used.

State equation of nonlinear system

$$x_k = f(x_{k-1}, U_k, 0)$$ (16)

Observation equation (or output equation)

$$Z_k = h(x_k, 0)$$ (17)

The other method for predicting, data prediction, is based on time series, including sliding window averaging and smoothing, has the following formula

$$\hat{x}_i = \frac{\sum_{j=1}^{k} \beta_j x_{i-j}}{k}$$ (18)

(K is the smooth sampling width)

| Method                | mean $|Y_{\text{pred}} - Y_{\text{real}}|_{\text{mean}}$ | max $|Y_{\text{pred}} - Y_{\text{real}}|_{\text{max}}$ | min $|Y_{\text{pred}} - Y_{\text{real}}|_{\text{min}}$ |
|----------------------|-----------------------------|-----------------------------|-----------------------------|
| Kalman filter        | 4.35                        | 11.23                       | 0.05                        |
| Time series smooth   | 28.72                       | 35.83                       | 11.65                       |

Result: The error obtained by using Kalman filter is lower than that of time series smoothing.

4. Security Analysis and Solutions

We propose four anti crawling [8] measures as follows.

1) The crawler have properties in headers, such as host, referer, user agent, cookie, etc. By analyzing system log, a single IP generates many requests in short time, determined as a crawler, and its IP is blocked.

2) Set the verification code picture, slide puzzle, SMS verification code, object recognition, problem selection, calculation results, etc. can use interval verification to verify the client.

3) For Phantom JS crawler, we can recognize the webdriver attribute through HTML JS and block IP. Ajax technology can also be used to load the website data asynchronously. The data loaded asynchronously will not be returned to HTML and cannot be crawled.
4) MD5, Sha and other encryption and coding algorithms are used for anti-crawling, and strong password can be set to prevent violent cracking and other hacker attacks.

5. Conclusion
For the point cloud data, this paper proposes cloth simulation filtering algorithm to separate the ground points [1,2], then carries out bilateral statistical filtering [2] to denoise, uses improved K-means clustering [3,4] to realize classification more precisely, and Kalman [5-7] filtering to predict the target and improve the accuracy. Finally, anti-crawler strategy [8] is proposed to ensure the security of information storage and transmission.

Acknowledgments
This paper is one of the phased achievements of the project "development of locomotive running gear condition monitoring and fault diagnosis system". Thanks to my mentor Luo Qian for her guidance.

References
[1] Deng Fei, dou Aixia, Wang Xiaoqing. Point cloud extraction of airborne lidar buildings after earthquake based on aerial images [J]. Acta remote sensing, 2017, 22 (S1)
[2] Li Pengfei, Wu hai, Jing Junfeng, et al. Noise classification and denoising algorithm for point cloud model [J]. Computer engineering and applications, 2016, 52 (020): 188-192
[3] Sun Zhongzhen. Extraction of glacier snow line in Qinghai Tibet Plateau Based on airborne lidar [D]. Chongqing Jiaotong University. 2019
[4] Xiong Si, Li Lei Min, Huang Yu Qing. Unstructured road detection based on wavelet transform and K-means [J]. Computer Engineering, 2014, 40 (2): 158-158
[5] Phuong T T , Ohishi K , Mitsantisuk C , et al. Disturbance Observer and Kalman Filter Based Motion Control Realization[J]. Ieej Journal of Industry Applications, 2018, 7(1):1-14.
[6] Li Jin. The adaptive tracking method of multi-dimensional feature point set in target decoration image [J]. Mechanical design and manufacturing engineering, 2019, 048 (007): 96-99
[7] Hongtao m, Saimin W. CAMSHIFT target tracking algorithm based on Kalman filtering [J]. Science and technology wind, 2019
[8] Li Pei. Research on web crawler and anti crawler technology based on Python [J]. Computer and digital engineering, 2019, 47 (6)