

Channel Extraction and Geometric Parameters Measurement Based on Point Clouds

Qingguo Zhang¹, Xiaolong Li¹, Huifang Feng^{1,*}, Jian Zhong¹, Yuehui Li¹, Michael A. Chapman², Jonathan Li³

¹ School of Computer and Software Engineering, Xihua University, 610039, Chengdu, China

² Department of Civil Engineering, Toronto Metropolitan University, 350 Victoria Street, Toronto, Ontario M5B 2K3, Canada

³ Department of Geography and Environmental Management & Department of Systems Design Engineering,
University of Waterloo, Waterloo, Ontario N2L 3G1, Canada

Keywords: Pre-embedded channel, Total station, Point clouds, Geometric parameters measurement, Gaussian distribution.

Abstract

In electrified railways, accurately measuring geometric parameters of pre-embedded tunnel channels is crucial for ensuring the installation precision and operational stability of the railway overhead contact network system. However, traditional manual measurement methods face challenges such as complex construction environments, stringent precision demands, and limited technical capabilities, resulting in inefficiency, significant safety risks, and poor repeatability. To address these issues, this paper introduces a novel framework based on total station point clouds. The framework comprises three key modules: point cloud preprocessing, channel extraction, and geometric parameter measurement. During preprocessing, point clouds are aligned using principal component analysis (PCA), and interference points are removed to enhance data quality. For channel extraction, an Otsu-based curvature threshold is first applied to preliminarily identify channel point clouds. Subsequently, a refined extraction process combining statistical denoising and density-based clustering is employed to isolate the channel point clouds with greater precision. In terms of geometric parameter measurement, an arc-based method is utilized for length measurement, while a generalized Gaussian distribution (GGD)-based approach is adopted for depth estimation. Experimental results demonstrate that the proposed method significantly improves channel extraction performance, achieving an F1-score improvement of up to 24.8%. Furthermore, the framework enables millimeter-level depth estimation with a mean absolute error of 1.90 mm.

1. Introduction

In the construction of tunnels for electrified railways, pre-embedded channels are an important facility in the catenary engineering. They are used to fix the catenary suspension columns, wire supports, and other equipment. These pre-embedded channels are usually constructed synchronously during the construction of the tunnel's secondary lining. However, due to various factors in the construction process, defects may occur in them, which in turn will affect the subsequent construction. It is a significant challenge to ensure the precision and quality of these pre-embedded channel systems due to the intricate tunnel environment and stringent construction requirements.

Conventionally, the method of manual measurement with the help of trolleys is the most widely used one. This requires personnel to undertake work at elevated heights, which has the disadvantages of low efficiency and high risk. With the development of computer vision, methods based on image processing technology have largely addressed the issue of high risk and have improved efficiency and accuracy to a certain extent (Archana and Jeevaraj, 2024). However, they are still limited by the lack of comprehensive spatial information and perform poorly in complex tunnel environments (Feng et al., 2021).

With the extensive utilization of total stations in the field of architecture, this paper proposes a novel framework. Its prominent characteristic is the ability to automatically extract the channel point clouds from the entire tunnel point cloud and accurately measure geometric parameters such as length and depth.

The framework includes three crucial modules: Point Clouds Preprocessing, Channel Extraction, and Geometric Parameters Measurement. This proposed framework exhibits outstanding performance in both the extraction of channels and the measurement of geometric parameters. The contributions of this study are outlined as follows:

- We introduce an integrated curvature analysis and Otsu-based threshold selection technique for channel extraction.
- The method encompassing statistical denoising and density-based clustering is applied to the point cloud to optimize the extraction effect.
- Employing the generalized Gaussian distribution yields reliable depth estimation outcomes in intricate and challenging complex environments.
- Experimental findings convincingly illustrate that the algorithm put forward in this study exhibits outstanding performance in both the extraction of channels and the estimation of geometric parameters.

2. Related Work

In recent years, point clouds have been extensively utilized in tasks encompassing localization (Li et al., 2023) (Li et al., 2024b), detection (Feng et al., 2021) (Feng et al., 2024), and reconstruction (Tang and Wang, 2024) (Li et al., 2024a) across a wide array of fields, and substantial accomplishments have been achieved. Additionally, with the continuous emergence of the demands for automation and intelligence in the construction of electrified railways, many studies have initiated the attempt

* Corresponding author: Huifang Feng (e-mail: fhf@xhu.edu.cn).

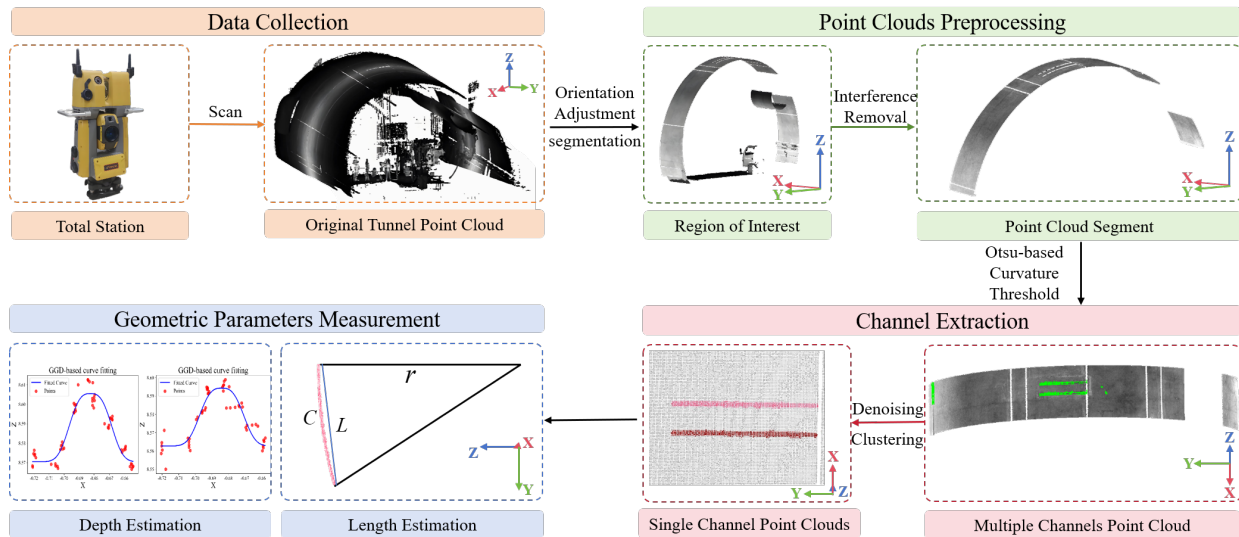


Figure 1. Workflow of the proposed method, consisting of three modules: Point Clouds Preprocessing, Channel Extraction, and Geometric Parameters Measurement.

to apply point clouds to detect railway tunnels. These studies can be broadly categorized into three main approaches.

The first category focuses on methods that attempt to obtain depth information from 2D images for 3D modeling. For instance, Lee et al. (2015) introduced a novel 3D model-based depth estimation method, which can effectively generate high-quality depth information for rigid objects during 2D to 3D stereoscopic conversion. However, such methods encounter significant challenges in practical applications. The generalization ability of the models often falls short, and computational efficiency remains a major issue, limiting their widespread use.

The second group of methods involves dimensionality reduction of 3D data. A cylinder fitting method (Duan et al., 2021) was proposed for high-density incomplete cylindrical point cloud data, generating a 2-D unwrapped depth map to reconstruct shield tunnel lining. Liu et al. (2022) developed a ResNet-based model using MLS point cloud images to detect water leakages in subway tunnels. Cui et al. (2024) unfolded 3D point clouds into 2.5D to obtain images and then used deep learning semantic segmentation networks designed for images to identify tunnel channels. While these dimensionality reduction techniques simplify point cloud processing, they inevitably cause the loss of depth information. This makes it difficult to accurately estimate the depth of channels, which is crucial for railway tunnel construction and maintenance.

The third category consists of methods that directly process point clouds. Huynh et al. (2021) utilized sparse point cloud data for depth estimation. SPCNet (Ji et al., 2023), a semi-supervised learning-based point cloud network, was designed for the segmentation of 3D tunnel scenes. It leveraged unlabeled point clouds to enhance segmentation performance while reducing the labeling burden. Nevertheless, these direct-processing methods have not fully exploited the spatial coordinates and feature information inherent in point clouds, leaving room for improvement in terms of accuracy and efficiency.

Although notable advancements have been achieved in tunnel crack detection and channel point cloud collection, current methodologies still struggle to integrally extract and precisely

estimate the depth of pre-embedded channels. Hence, the development of an approach capable of maximally leveraging 3D point cloud data holds immense significance for enhancing the accuracy and efficiency of railway tunnel channel inspection.

3. Method

This paper presents an extraction and geometric parameters measurement method for the pre-embedded channels in electrified railway tunnels based on point clouds. The workflow of the proposed method is illustrated in Figure 1.

3.1 Point Clouds Preprocessing

3.1.1 Orientation Adjustment: Due to the different installation positions of the total station, the orientation of the original point cloud is misaligned with the world coordinate system (WCS). Meanwhile, the direction of the pre-embedded channels is closely related to the direction of tunnel progression, and it is usually perpendicular. Therefore, adjusting the direction of the tunnel point cloud is extremely necessary. This adjustment ensures that the direction of tunnel progression is parallel to the x-axis of the WCS. To determine the tunnel's progression direction, we apply Principal Component Analysis (Abdi and Williams, 2010).

Specifically, we select a small portion of the point cloud from the tunnel roof according to the z-coordinates of the points, and apply PCA to it. The eigenvector with the largest eigenvalue is identified as the principal direction, representing the tunnel's progression. Then, we calculate the angle between this direction and the WCS x-axis, and rotate the original point cloud to align it parallel to the x-axis. This alignment greatly facilitates subsequent processing and geometric parameters measurement. Based on prior knowledge, the effective scanning distance of the total station is considerably greater than the distance from the region of interest to the total station. To reduce the time and space required for subsequent processing, we use the scanner position as the center and set a bounding box in the xy plane. Then, point cloud within the x-axis interval $[-x, x]$ are retained.

3.1.2 Interference Removal: Owing to the characteristics of the panoramic scanner, the scanner is capable of capturing all the spatial information within its scanning range, including the tunnel wall where the channels are situated and the objects inside the tunnel, such as people, vehicles, ventilation duct, and other facilities, as shown in Figure 2 (a). These interferences inside the tunnel will significantly affect the accuracy of channel extraction. Consequently, this paper proposes a rapid and efficient point cloud preprocessing method to remove the interferences inside the tunnel.

For some interferences with relatively low elevation, such as the pavement, people, vehicles, etc., a height threshold H_1 is established to filter out points whose z-coordinates are lower than H_1 . In this way, these interferences can be effectively removed, resulting in a point cloud $P = \{p_1, p_2, \dots, p_n\}$, and $p_i = (x_i, y_i, z_i)$, where n denotes the total number of points.

For the ventilation duct, it is usually a cylindrical structure affixed to the tunnel wall at a high elevation. Overall, the point cloud P approximates a semi-cylindrical surface. Consequently, a method of filtering the point cloud by fitting a cylindrical surface is proposed. Specifically, we retain the part of the point cloud positioned at the top of the tunnel, whose z-coordinates are usually greater than H_2 . In this way, the influence of the ventilation duct on the fitting effect can be avoided. Then, we project this part of the point cloud onto the yo z plane to obtain a set of two-dimensional scattered points distributed in an arc shape, and use the least squares method to determine the radius r and the center coordinates (y_c, z_c) of the optimal fitting circle for these points. Since the extension direction of the tunnel point cloud P is consistent with the x-axis direction, r is also the radius of the cylindrical surface, and the straight line passing through the point $(0, y_c, z_c)$ and parallel to the x-axis is the axis l of the cylindrical surface, and its equation is as follows:

$$\begin{cases} y = y_c \\ z = z_c \end{cases} \quad (1)$$

Meanwhile, for $\forall p_i \in P$, $p_i = (x_i, y_i, z_i)$, $i \in \{1, 2, \dots, n\}$, calculate the Euclidean distance d_i from p_i to the axis line l of the cylindrical surface and Δ_i as follows:

$$d_i = \sqrt{(y_i - y_c)^2 + (z_i - z_c)^2}, \quad (2)$$

$$\Delta_i = |d_i - r|. \quad (3)$$

Due to the characteristic of the laser propagating in a straight line, one side of a ventilation duct is captured in the point cloud, and it will block the adjacent tunnel wall, creating a gap. Based on this gap, we establish a tolerance value t , which determines the maximum allowable deviation of the distance from point p_i to the cylindrical surface. As shown in Figure 2 (b), we retain the points for which $\Delta_i \leq t$, resulting in a refined tunnel point cloud $P' = \{p'_1, p'_2, \dots, p'_m\}$, where m is the number of points in P' .

3.2 Channel Extraction

The curvature is an important metric for characterizing the local geometric features of each point in the point cloud. It reflects the degree of surface variation at that point, namely the surface

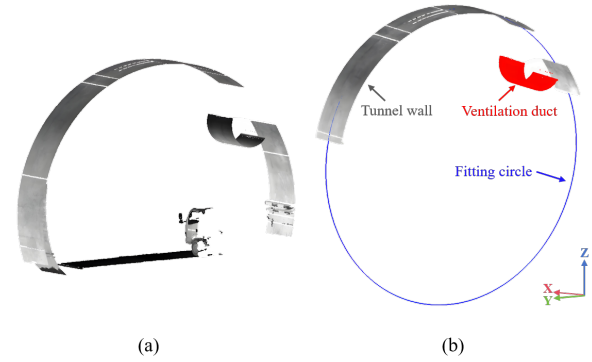


Figure 2. Illustration of the removal of interferences in the tunnel. (a) The point cloud Segment of the tunnel containing a vehicle and a ventilation duct. (b) Removal of the ventilation duct through the circle fitting.

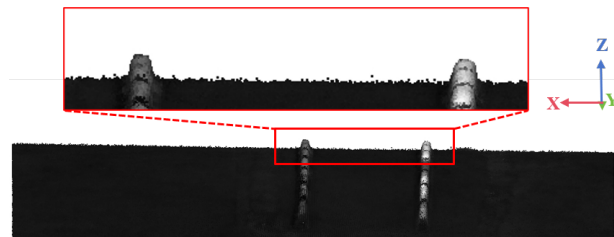


Figure 3. Illustration of the differences in shape between channels and the tunnel wall.

curvature degree. The tunnel point cloud shows apparent differences in shape between channels and the tunnel wall. As shown in Figure 3, areas of channels exhibit noticeable protrusions, whereas areas of the tunnel wall without channels are relatively smooth. By leveraging the prior knowledge of channel design specifications, analyzing the characteristics of curvature can aid in the extraction of the channels.

This paper proposes a novel method for extracting channel point clouds. Channel point clouds are extracted by the selection of the curvature threshold, which is based on the Otsu algorithm. Moreover, multi-region extraction in complex scenarios is optimized via a statistical-based denoising algorithm (Carrilho et al., 2018) and a density-based clustering algorithm (Han et al., 2020).

3.2.1 Calculating Curvatures: Since the acquired point cloud is extremely dense, voxel downsampling is carried out on the point cloud to balance computational efficiency and the preservation of geometric features. Specifically, the space where the point cloud P' lies is divided into cubes (voxels) with a size of V , and the voxel to which each point belongs is ascertained according to its coordinates. For each non-empty voxel, the centroid of the voxel is selected as its representative point to form a new point cloud P'' . Through this voxel downsampling approach, the data volume is substantially decreased. Meanwhile, the prevention of local feature loss is ensured.

For the calculation of curvature, we employ a method based on k-nearest neighbor (KNN) and covariance matrix decomposition. Specifically, for each point p''_i in the point cloud P'' , we utilize the KNN to identify its k nearest neighbor points $\{p''_{i_1}, p''_{i_2}, \dots, p''_{i_k}\}$. Then, we normalize these points

to obtain the neighborhood point coordinate matrix $\mathbf{X}_i = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k\}$, where $\mathbf{x}_j = \mathbf{p}_{i_j}'' - \mathbf{p}_i''$. Calculating the covariance matrix $\mathbf{C}_i$ of $\mathbf{X}_i$ as follows:

$$\mathbf{C}_i = \frac{1}{k-1} \sum_{j=1}^k (\mathbf{x}_j - \bar{\mathbf{x}})(\mathbf{x}_j - \bar{\mathbf{x}})^T, \quad (4)$$

where $\bar{\mathbf{x}}$ is the mean of the neighborhood points. However, since the coordinates have been normalized, $\bar{\mathbf{x}} \approx 0$, it is simplified to

$$\mathbf{C}_i \approx \frac{1}{k} \sum_{j=1}^k \mathbf{x}_j \mathbf{x}_j^T. \quad (5)$$

$\mathbf{C}_i$ is a symmetric matrix, which characterizes the distribution characteristics of the neighborhood points in the three-dimensional space. The eigenvalues $\lambda_1, \lambda_2, \lambda_3$ can be derived by performing the eigenvalue decomposition on $\mathbf{C}_i$ and the curvatures δ_i can be calculated as follows:

$$\mathbf{C}_i = \mathbf{V} \cdot \mathbf{\Lambda} \cdot \mathbf{V}^T, \quad (6)$$

$$\delta_i = \frac{\lambda_1}{\lambda_1 + \lambda_2 + \lambda_3 + \epsilon}, \quad (7)$$

where $\mathbf{\Lambda} = \text{diag}(\lambda_1, \lambda_2, \lambda_3)$, $\lambda_1 \leq \lambda_2 \leq \lambda_3$ and ϵ is added to prevent division-by-zero errors.

3.2.2 Otsu-based Curvature Threshold Selection: To achieve the separation of the channel points and the tunnel wall points, we employed the Otsu algorithm. First, we excluded those invalid points whose curvature is less than or equal to 0, and normalized the valid points to the range of [0, 1] as follows:

$$\mathcal{C}_{\text{valid}} = \{\delta_i \mid \delta_i > 0, \forall \mathbf{p}_{i_j}'' \in P''\}, \quad (8)$$

$$\delta'_i = \frac{\delta_i - \min(\mathcal{C}_{\text{valid}})}{\max(\mathcal{C}_{\text{valid}}) - \min(\mathcal{C}_{\text{valid}})}. \quad (9)$$

Then, the Otsu algorithm was utilized to determine the threshold T^* that maximizes the between-class variance, and T^* was mapped back to the original curvature range through Equation 10 to obtain the threshold value, denoted as T . Finally, based on the T , the point cloud P'' was divided into the channel point cloud with high curvature and the tunnel wall point cloud with low curvature. The formulas are:

$$T = T^* \cdot (\max(\mathcal{C}_{\text{valid}}) - \min(\mathcal{C}_{\text{valid}})) + \min(\mathcal{C}_{\text{valid}}), \quad (10)$$

$$\begin{aligned} P_{\text{channels}} &= \{\mathbf{p}_i'' \mid \delta_i > T\}, \\ P_{\text{wall}} &= \{\mathbf{p}_i'' \mid \delta_i \leq T\}. \end{aligned} \quad (11)$$

3.2.3 Denoising and Clustering: The method based on Otsu's curvature threshold can retain the channel points relatively well. However, due to the complex situations at the construction site, there might exist cavities or other additional facilities on the tunnel wall. The points in these areas may possess

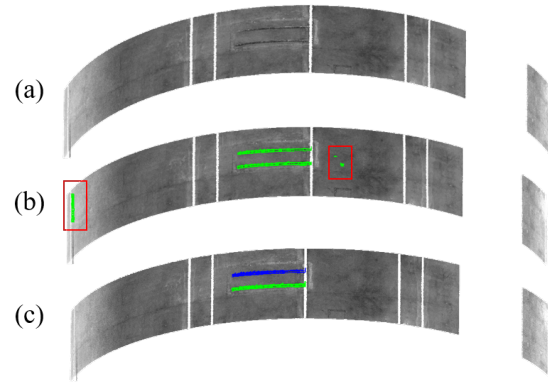


Figure 4. Illustration of the channel extraction. (a) The point cloud segment of interest. (b) The channel extraction results of the Otsu-based curvature threshold method. The red rectangular area indicates the false positive point. (c) The result of channel clustering. The blue points and the green points signify two channels, respectively.

curvature characteristics similar to those of the channel points, and thus cannot be removed relying solely on the curvature. Moreover, the distribution of these points is quite random. If they are not removed, they will directly affect the results of subsequent geometric parameters measurement.

The core idea of the statistical-based denoising algorithm (Carvalho et al., 2018) is to process data based on local statistical characteristics, to remove noise or extract useful information. This approach mainly relies on the relationships between each point and its neighboring points. Specifically, for each point p_i in the point cloud, calculate its average distance to the nearest $nb_neighbors$ points:

$$d_i = \frac{1}{k} \sum_{j=1}^k \|p_i - q_j\|_2. \quad (12)$$

Subsequently, under the assumption that the neighborhood distances of all points adhere to a Gaussian distribution, compute the global mean value μ and the standard deviation σ .

$$\mu = \frac{1}{N} \sum_{i=1}^N d_i, \quad (13)$$

$$\sigma = \sqrt{\frac{1}{N-1} \sum_{i=1}^N (d_i - \mu)^2}, \quad (14)$$

$$\tau = \mu + std_ratio \times \sigma, \quad (15)$$

where std_ratio is an adjustable parameter. Outlier points are identified by setting a specific threshold value τ . Points that meet the condition $d_i \leq \tau$ are retained, thereby yielding the filtered point cloud of the channels denoted as P'_{channels} .

Since the point cloud of the channels may contain multiple channels, it is necessary to divide them into point clouds that contain only a single channel to ensure the accuracy of the measurement. To achieve this objective, we employed the DBSCAN algorithm to conduct a clustering analysis on the point

cloud of multiple channels. DBSCAN, namely Density-Based Spatial Clustering of Applications with Noise, is a classical density-based clustering algorithm. It possesses the remarkable capability to detect clusters of diverse and arbitrary shapes without the need for prior determination of the exact number of clusters. Moreover, it is proficient in discerning noise points within the dataset.

DBSCAN involves four key concepts: Core Point, Directly Density-Reachable, Density-Reachable, and Density-Connected.

- **Core Point:** For a given point p , if there are at least $MinPts$ points (including itself) within its neighborhood with a radius of ϵ , then this point is called a core point.

$$N_\epsilon(p) = \{q \in D | dist(p, q) \leq \epsilon\}, \quad (16)$$

where $N_\epsilon(p)$ represents the set of all points within the ϵ -neighborhood of the point p , and $dist(p, q)$ denotes the distance separating the two points p and q .

- **Directly Density-Reachable:** A point q is considered to be directly density-reachable from a point p if and only if p is a core point and q lies within the ϵ -neighborhood of p .
- **Density-Reachable:** A point q is considered to be density-reachable from a point p if there exists a sequence of points $p_1, p_2, \dots, p_n$, where $p_1 = p$, $p_n = q$, and for all $i < n$, the point p_{i+1} is directly density-reachable from p_i .
- **Density-Connected:** Two points p and q are considered to be density-connected if there exists a point o such that both p and q are density-reachable from o .

Specifically, first, initialize all the points within the point cloud $P'_{channels}$ as unvisited points. Then, determine whether a point is a core point. If a point $p'_{channels}$ does not satisfy the condition $|N_\epsilon(p'_{channels})| \geq MinPts$, mark it as a noise point. Otherwise, create a sub-point cloud, and add the point $p'_{channels}$ and all the points that are directly density-reachable within its ϵ -neighborhood to this sub-point cloud P'_{single} . For each newly added point, repeat the above steps until there are no further new points that can be incorporated. In this way, a sub-point cloud P'_{single} for a single channel can be obtained. Subsequently, continue to process the remaining unvisited points until all points have been visited and all sub-point clouds have been completely constructed. As shown in Figure 4, through the DBSCAN algorithm, a series of sub-point clouds for a single channel will be obtained.

3.3 Geometric Parameters Measurement

3.3.1 Arc-based Length Estimation: Since the point cloud of the channel is distributed in an arc shape, this paper adopts an arc-based method to measure the length of the channel. Specifically, the point with the minimum y-coordinate and the point with the maximum y-coordinate in P'_{single} are found, and the chord length L is calculated as follows:

$$L = \sqrt{(y_{min} - y_{max})^2 + (z_{min} - z_{max})^2} \quad (17)$$

Then, according to and the radius r obtained in Section 3.1.2, the arc length C is calculated as follows:

$$C = 2r \cdot \arcsin\left(\frac{L}{2r}\right) \quad (18)$$

Parameter	Value
Height threshold (H_1)	3.5m
Height threshold (H_2)	7.0m
Tolerance (t)	0.2m
Voxel size (V)	0.015m
Neighbor points (k)	50
Small offset (ϵ)	1×10^{-8}
Statistical filtering ($nb_neighbors$)	15
Adjustable parameter (std_ratio)	1.5
Least points ($MinPts$)	100
Neighbor radius (ϵ)	0.05m
Segmentation angle (θ_{seg})	10°
Step angle (θ_{stp})	3°
Shape parameter (β)	2

Table 1. Geometric Parameter Measuring.

3.3.2 GGD-based Depth Estimation: To rapidly obtain the cross-section of the channel, a radian-based method is used to segment the channel point cloud. Specifically, based on the center coordinates (y_c, z_c) obtained in Section 3.1.2, the channel point cloud P'_{single} is segmented according to the preset segmentation angle θ_{seg} and step angle θ_{stp} , thereby obtaining multiple sub-point clouds $P'_{single}^1, P'_{single}^2, \dots, P'_{single}^s$, s is determined by the length of the channel and θ_{stp} .

To achieve precise estimation of the channel depth, a depth estimation algorithm based on the Generalized Gaussian Distribution is proposed. Specifically, for a P'_{single}^s , it is projected onto the cross-section plane to achieve dimensionality reduction. As a result, the two-dimensional scatter points of the projected channel are obtained and denoted as Q . Then, an algorithm utilizing the GGD (Novey et al., 2010) is employed to fit the curve of these scatter points. The depth of the channel is calculated based on the results of this curve fitting. This method significantly enhances the accuracy and reliability of channel depth estimation.

For each $q_i \in Q$, $q_i = (x_i, z_i)$, the formulation of GGD is given by:

$$f(q_i, \alpha, \beta, \mu) = \frac{\beta}{2\alpha\Gamma\left(\frac{1}{\beta}\right)} e\left(\left|\frac{q_i - \mu}{\alpha}\right|\right)^\beta, \quad (19)$$

where μ is the mean coordinates of Q , α represents the scale parameter analogous to the standard deviation, β represents the shape parameter influencing the form of the distribution, and Γ denotes the gamma function involved in the normalization process of the distribution.

4. Experiment and analysis

4.1 Data Description

In this study, we utilized the Topcon GTL-1200 total station to acquire data from the electrified railway tunnel that was under construction for our experiments. The laser scanning unit of the Topcon GTL-1200 has a maximum measurable range of approximately 70 meters. It is capable of emitting 200,000 points per second and attaining an accuracy of up to 2 millimeters. Our dataset encompasses 36 railway tunnels. Each tunnel's point cloud data has dimensions of roughly 110 meters in length, 10 meters in width, and 10 meters in height. On average, each tunnel consists of approximately 16 million points, with the minimum distance between adjacent points being 0.2 millimeters.

For this experiment, we collected 36 tunnel point clouds and randomly selected 15 tunnel point clouds with 30 channels.

T	Fixed T			Otsu-based T			Otsu-based T + Denoising & Clustering		
	Precision	Recall	F1-score	Precision	Recall	F1-score	Precision	Recall	F1-score
1	82.02	70.56	75.86	95.46	83.74	89.22	97.90	86.13	91.64
2	60.57	99.81	75.39	87.33	85.75	86.53	90.99	89.02	89.99
3	85.91	60.87	71.25	82.93	88.29	85.53	86.56	91.74	89.07
4	56.13	99.92	71.88	75.38	89.16	81.69	80.00	91.25	85.26
5	59.72	99.94	74.76	77.82	98.72	87.03	81.02	93.06	86.63
6	96.46	75.05	84.42	78.87	89.36	83.79	83.27	94.88	88.70
7	97.18	72.48	83.03	79.49	79.28	79.38	75.47	92.17	82.99
8	88.23	53.64	66.72	84.29	94.67	89.18	89.11	88.60	88.85
9	87.84	62.21	72.84	86.64	96.50	91.30	88.36	90.70	89.51
10	83.01	77.74	80.29	88.14	82.44	85.19	91.31	85.46	88.29
11	88.41	72.61	79.73	91.33	91.21	91.27	93.64	90.20	91.89
12	92.64	73.00	81.66	93.74	91.76	92.74	95.16	91.14	93.11
13	89.68	67.33	76.91	88.53	87.82	88.17	89.92	97.26	93.44
14	87.09	77.19	81.84	86.61	97.73	91.83	87.20	97.16	91.91
15	96.14	50.68	66.37	93.41	85.50	89.28	92.30	90.05	91.17

Table 2. Comparison of the channel extraction results (%) of different methods.

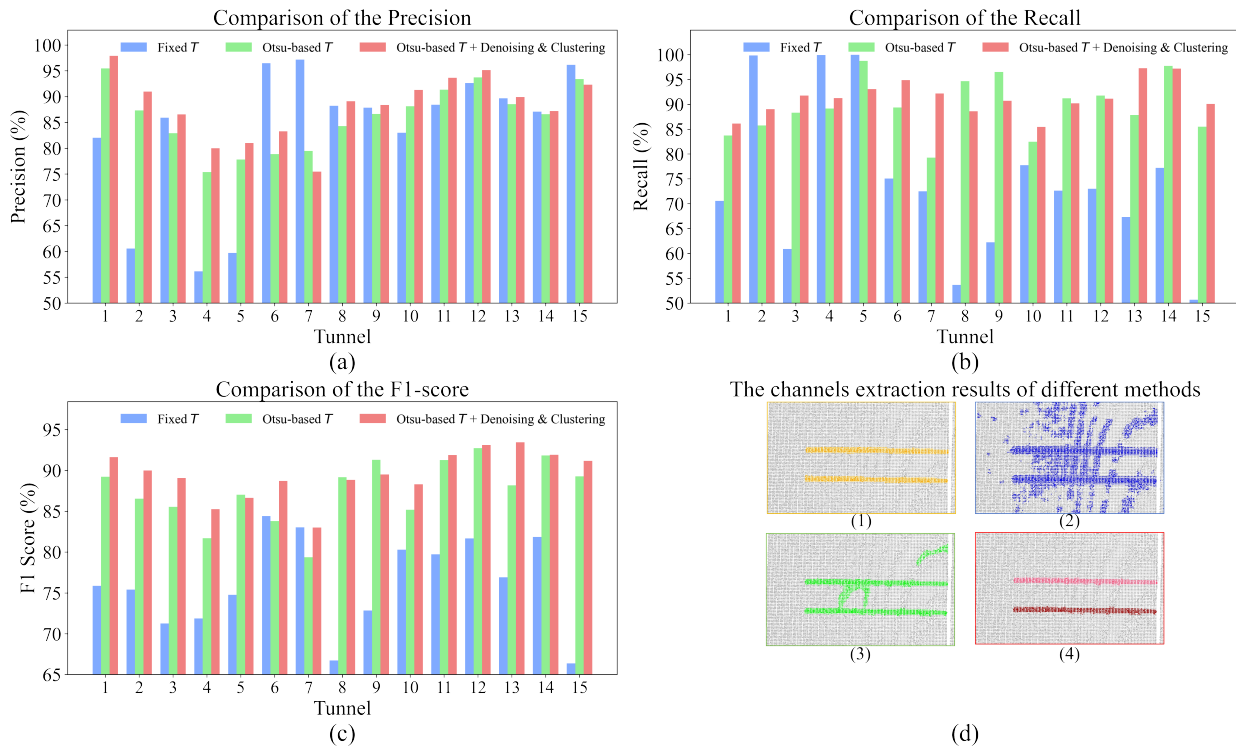


Figure 5. Illustration of the comparison of the channel extraction results of different methods. (a), (b), and (c) show the comparisons of different methods regarding precision, recall, and F1-score, respectively. (d) shows the comparison of the channel extraction results in Tunnel No. 13. (1) The ground truth. (2) The method is based on a fixed curvature threshold. (3) The Otsu-based curvature threshold method. (4) The method that integrates the Otsu-based curvature threshold with denoising and clustering.

These tunnels were manually annotated as the ground truth of the Channels extraction. Due to the difficulties in manually measuring geometric parameters at the construction site, for the length, we used the design standard values of the channels as the ground truth. For the depth, instead of conducting manual measurements on-site, we used the manual measurement data obtained from the 3D point cloud as the ground truth. This meticulous and stringent approach guarantees both the precision and the validity of our results.

4.2 Implementation

For this paper, our algorithms have been implemented in PyCharm 2020.1 and run on a PC with Intel Core i7-1165G7 @ 2.8GHz, 16GB RAM. The parameter setting is listed in Table 1.

4.3 Ablation Experiments

Regarding the extraction of channels, we carried out three distinct experiments: channel extraction utilizing a fixed curvature threshold, employing the Otsu-based curvature threshold, and applying the integration of the Otsu-based curvature threshold with denoising and clustering. As illustrated in Table 2, we compared the precision, recall, and F1-score to showcase the contribution of our designed architectures to the effectiveness of channel extraction.

As shown in Figure 5, for the method relying on the fixed threshold, there are two extreme scenarios. In Tunnel No. 4, the method exhibits a recall of 99.92% but a precision of only 56.13%, where the extracted channel point cloud contains sub-

Channel	Length			Depth							
	Length GT	<i>Ours</i>	$diff_1$	Depth GT	<i>V</i>	$diff_2$	<i>Ours_P1</i>	<i>Ours_P2</i>	<i>Ours_P3</i>	<i>Ours_Avg</i>	$diff_3$
1-1	2000.00	1989.39	10.61	31.16	35.73	4.57	31.57	34.77	32.60	32.98	1.82
1-2	2000.00	1980.30	19.70	31.72	33.35	1.63	32.82	31.54	32.29	32.22	0.50
2-1	2000.00	1977.88	22.12	28.91	33.84	4.93	29.97	31.98	30.76	30.90	1.99
2-2	2000.00	1968.01	31.99	28.80	29.49	0.69	28.14	32.69	28.20	29.68	0.88
3-1	2000.00	1977.59	22.41	26.72	35.75	9.03	28.65	25.40	26.04	26.70	0.02
3-2	2000.00	1970.12	29.88	27.70	33.84	6.14	28.39	30.98	26.37	28.58	0.88
4-1	2000.00	1979.28	20.72	28.03	31.57	3.54	30.77	31.77	29.61	30.72	2.69
4-2	2000.00	1965.37	34.63	27.99	29.49	1.50	30.15	30.85	26.68	29.23	1.24
5-1	2000.00	1987.21	12.79	29.29	36.23	6.94	29.75	25.69	26.68	27.37	1.92
5-2	2000.00	1962.84	37.16	27.93	34.37	6.44	30.79	25.89	27.94	28.21	0.28
6-1	2000.00	1965.18	34.82	30.69	27.51	3.18	32.76	34.28	37.47	34.84	4.15
6-2	2000.00	1972.44	27.56	31.08	26.82	4.26	32.55	29.90	29.27	30.57	0.51
7-1	2000.00	1963.22	36.78	29.28	42.03	12.75	30.90	39.16	35.04	35.03	5.75
7-2	2000.00	1963.15	36.85	28.98	27.51	1.47	32.81	33.15	36.18	34.05	5.07
8-1	2000.00	1990.45	9.55	29.48	34.66	5.18	32.76	30.14	29.32	30.74	1.26
8-2	2000.00	1973.40	26.60	28.81	32.32	3.51	31.78	29.52	28.89	30.06	1.25
9-1	2000.00	1962.41	37.59	30.08	29.67	0.41	28.53	30.93	29.36	29.61	0.47
9-2	2000.00	1975.11	24.89	28.89	22.06	6.83	27.62	28.80	26.79	27.74	1.15
10-1	2000.00	1962.11	37.89	28.50	40.24	11.74	29.65	31.84	29.16	30.22	1.72
10-2	2000.00	1974.93	25.07	27.03	34.42	7.39	28.61	30.19	28.52	29.11	2.08
11-1	3500.00	3503.58	3.58	28.00	41.36	13.36	32.22	35.37	32.91	33.50	5.50
11-2	3500.00	3515.79	15.79	27.35	36.40	9.05	33.82	31.85	27.67	31.11	3.76
12-1	3500.00	3514.11	14.11	29.30	36.05	6.75	30.58	33.08	33.00	32.22	2.92
12-2	3500.00	3507.48	7.48	28.15	35.99	7.84	28.66	32.34	31.91	30.97	2.82
13-1	3500.00	3503.40	3.40	27.76	21.54	6.22	27.28	27.00	27.90	27.39	0.37
13-2	3500.00	3504.98	4.98	27.66	29.97	2.31	29.29	27.80	27.04	28.04	0.38
14-1	3500.00	3503.82	3.82	26.13	29.97	3.84	28.20	27.54	27.91	27.88	1.75
14-2	3500.00	3498.71	1.29	28.29	33.81	5.52	31.48	28.26	27.96	29.23	0.94
15-1	2000.00	1962.42	37.58	27.54	35.91	8.37	29.52	30.65	27.71	29.29	1.75
15-2	2000.00	1958.52	41.48	29.56	22.14	7.42	32.66	28.54	31.15	30.78	1.22
Avg			22.44	28.69	32.47	5.76				30.30	1.90
S			12.60		5.14	3.32				2.19	1.54

Table 3. Experimental results of geometric parameters measurement. *V*: The depth measured by the RANSAC-based method (Wang et al., 2023). $diff_1$: The absolute difference of length *Ours* and Length GT. $diff_2$: The absolute difference of depth *V* and Depth GT. $diff_3$: The absolute difference of depth *Ours_Avg* and Depth GT.

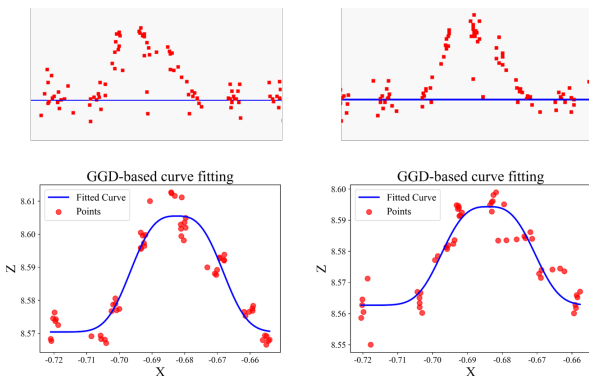


Figure 6. Fitting results of channel segment point clouds: method (Wang et al., 2023) (first row), our method (second row).

stantial noise points. In contrast, for Tunnel No. 7, the method achieves a precision of 97.18% but a recall of only 72.48%, with the structural gaps in the extracted point cloud. These results lead to lower F1-scores, revealing that a fixed curvature threshold is incapable of adeptly dealing with diverse tunnel point clouds. The Otsu-based curvature threshold mitigates this issue, while the approach combining statistical denoising and density-based clustering further enhances channel extraction performance, achieving an F1-score improvement of up to 24.8%.

4.4 Comparative Experiments

In terms of the measurement of geometric parameters, our method yields accurate and consistent results as shown in Table 3.

Regarding the length measurement, the maximum percentage

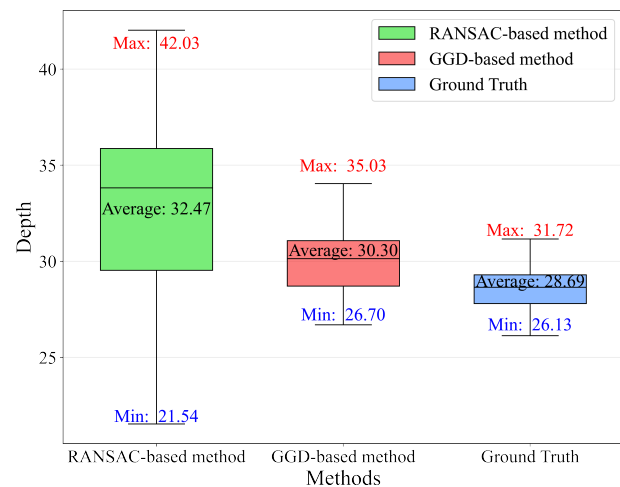


Figure 7. Illustration of the depth distributions of the RANSAC-based method (Wang et al., 2023) and the GGD-based method.

of absolute error between the measured length and the ground truth is 2.074%, with an average of 22.44 mm and a standard deviation of 12.60 mm. For depth measurement, as shown in Figure 6, we employed the GGD-based method to select three sub-point clouds from the point cloud of a single channel for measurement, obtaining *Ours_P1*, *Ours_P2*, *Ours_P3*, and then calculated their average values *Ours_Avg*. As shown in Figure 7, the average values of depth measurements by our method ranged from 26.70 mm to 35.03 mm, with a mean of 30.30 mm and a standard deviation of 2.19 mm. Compared with the ground truth, the absolute errors ranged from 0.02 mm to 5.75 mm, with an average of 1.90 mm and a standard devi-

ation of 1.54 mm. In contrast, the depth measurement values of a plane fitting algorithm using RANSAC (Wang et al., 2023) ranged from 21.54 mm to 42.03 mm, with an average of 32.47 mm and a standard deviation of 5.14 mm. In addition, its absolute errors range from 0.41 mm to 13.36 mm, and the average and the standard deviation reach 5.76 mm and 3.32 mm, respectively.

From the above experiments, it can be observed that our method provides more accurate depth values. On the other hand, the standard deviation of the depth obtained by our method is significantly smaller than that of the method (Wang et al., 2023). This indicates that our method yields more reliable and consistent results, with smaller systematic error.

5. Conclusion

Pre-embedded channels play an extremely crucial role in electrified railway tunnels. They are the key to the efficient installation and operation of the railway overhead contact system. Measuring their geometric parameters is an essential task before installing relevant facilities. Based on the point cloud data collected from the total station, this paper proposes an automated framework that enables the extraction and measurement of the channels. Moreover, accurate and consistent results verified by experiments have been achieved for both processes. This framework provides an efficient and safe technology for inspection tasks. Future work will focus on enhancing the robustness of existing algorithms. Additionally, according to the emerging requirements of railway construction, supplementary algorithms will be designed to enable the measurement of a broader spectrum of geometric parameters.

References

- Abdi, H., Williams, L. J., 2010. Principal component analysis. *Wiley interdisciplinary reviews: computational statistics*, 2(4), 433–459.
- Archana, R., Jeevaraj, P. E., 2024. Deep learning models for digital image processing: a review. *Artificial Intelligence Review*, 57(1), 11.
- Carrilho, A., Galo, M., Santos, R., 2018. Statistical outlier detection method for airborne LiDAR data. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 42, 87–92.
- Cui, H., Mao, Q., Li, J., Hu, Q., Tao, Y., Ma, J., Li, Z., 2024. Shield Tunnel Dislocation Detection Method Based on semantic segmentation and bolt hole positioning of MLS point cloud. *IEEE Transactions on Geoscience and Remote Sensing*, 62, 1–15.
- Duan, D., Qiu, W., Cheng, Y., Zheng, Y., Lu, F., 2021. Reconstruction of shield tunnel lining using point cloud. *Automation in Construction*, 130, 103860.
- Feng, H., Li, W., Luo, Z., Chen, Y., Fatholahi, S. N., Cheng, M., Wang, C., Junior, J. M., Li, J., 2021. GCN-based pavement crack detection using mobile LiDAR point clouds. *IEEE Transactions on Intelligent Transportation Systems*, 23(8), 11052–11061.
- Feng, H., Li, W., Ma, L., Chen, Y., Guan, H., Yu, Y., Marcato Junior, J., Li, J., 2024. Crack-U2Net: Multiscale Feature Learning Network for Pavement Crack Detection From Large-Scale MLS Point Clouds. *IEEE Transactions on Intelligent Transportation Systems*, 25(11), 17952–17964.
- Han, X., Armenakis, C., Jadidi, M., 2020. DbSCAN optimization for improving marine trajectory clustering and anomaly detection. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLIII-B4-2020, 455–461.
- Huynh, L., Nguyen, P., Matas, J., Rahtu, E., Heikkilä, J., 2021. Boosting monocular depth estimation with lightweight 3d point fusion. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 12767–12776.
- Ji, A., Zhou, Y., Zhang, L., Tiong, R. L., Xue, X., 2023. Semi-supervised learning-based point cloud network for segmentation of 3D tunnel scenes. *Automation in Construction*, 146, 104668.
- Lee, J., Kim, Y., Lee, S., Kim, B., Noh, J., 2015. High-quality depth estimation using an exemplar 3d model for stereo conversion. *IEEE transactions on visualization and computer graphics*, 21(7), 835–847.
- Li, Q., Feng, H., Shi, K., Gao, Y., Fang, Y., Liu, Y.-S., Han, Z., 2024a. Learning signed hyper surfaces for oriented point cloud normal estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12), 9957–9974.
- Li, W., Yang, Y., Yu, S., Hu, G., Wen, C., Cheng, M., Wang, C., 2024b. Diffloc: Diffusion model for outdoor lidar localization. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 15045–15054.
- Li, W., Yu, S., Wang, C., Hu, G., Shen, S., Wen, C., 2023. Sgloc: Scene geometry encoding for outdoor lidar localization. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9286–9295.
- Liu, S., Sun, H., Zhang, Z., Li, Y., Zhong, R., Li, J., Chen, S., 2022. A multiscale deep feature for the instance segmentation of water leakages in tunnel using MLS point cloud intensity images. *IEEE Transactions on Geoscience and Remote Sensing*, 60, 1–16.
- Novey, M., Adali, T., Roy, A., 2010. A complex generalized Gaussian distribution—Characterization, generation, and estimation. *IEEE Transactions on Signal Processing*, 58(3), 1427–1433.
- Tang, H., Wang, Z., 2024. Urban building coverage areas calculation and 3d modeling based on uav lidar. *IGARSS 2024-2024 IEEE International Geoscience and Remote Sensing Symposium*, 8638–8641.
- Wang, B., Zhao, Z., Chen, Y., Yu, J., 2023. A novel robust point cloud fitting algorithm based on nonlinear Gauss–Helmert model. *IEEE Transactions on Instrumentation and Measurement*, 72, 1–12.