# Training-free Semantic Segmentation of Shield Tunnel Point Clouds

Qiushi Wang[1,2,3], Yafei Qiao[1,2], Kourosh Khoshelham[3], Wenqi Ding[1,2]

[1] Department of Geotechnical Engineering, Tongji University, Shanghai, China - (2011129, yafei.qiao, dingwq)@tongji.edu.cn
[2] Key Laboratory of Geotechnical and Underground Engineering of Ministry of Education, Tongji University, Shanghai, China
[3] Department of Infrastructure Engineering, the University of Melbourne, Victoria, Australia - k.khoshelham@unimelb.edu.au

**Keywords:** Laser Scanning, Point Cloud Sampling, Non-parametric Network, Few-shot, Similarity based Learning

## Abstract

Semantic segmentation of shield tunnel point clouds provides valuable information for checking assembly quality, deformation, or defects. To achieve efficient semantic segmentation of shield tunnel point clouds, Tunnel-NN, a training-free non-parametric network is proposed. Tunnel-NN is evaluated using a dataset of five point clouds of shield tunnel rings and compared to two common baselines, namely PointNet and PointNet++, demonstrating comparable performance. To further enhance its performance in large-scale point cloud applications, a sector splitter sampling method is introduced based on tunnel section geometry. This approach preserves local geometric features while reducing the size of input data. Test results indicate that the sector splitter significantly improves the segmentation accuracy of Tunnel-NN and also benefits PointNet and PointNet++. Compared to trainable deep learning algorithms, Tunnel-NN achieves similar performance on datasets with few training examples without the need for training, highlighting its potential for broad engineering applications.

## 1. Introduction

The underground railway system has become a vital component of urban transportation, owing its utility to ensure extensive connectivity and interaction within the region, thus facilitating population growth and significantly contributing to regional economic development and social prosperity (Mejia-Dorantes et al., 2012, Peng et al., 2019). Currently, the shield tunneling method is widely employed in subway tunnel construction due to its high efficiency and minimal disruption. The lining structure of a shield tunnel consists of precast reinforced concrete segments, with joints between adjacent segments connected by bolts or other mechanisms, as illustrated in Figure 1. Despite the advantages in construction efficiency, offsets and openings at these joints are common during the assembly process (Zhu et al., 2014, Zhao et al., 2023, Li et al., 2024a). Over time, joint deformation can progressively worsen during tunnel operation and maintenance, leading to segment cracking, water seepage at the joints, and peeling of segment corners. These issues not only increase safety risks but also degrade long-term service performance and raise maintenance costs, thereby negatively impacting metro operations (Attard et al., 2018). Consequently, timely and effective inspections are critical to ensuring high assembly quality during construction and maintaining safety throughout the tunnel’s operational lifespan.

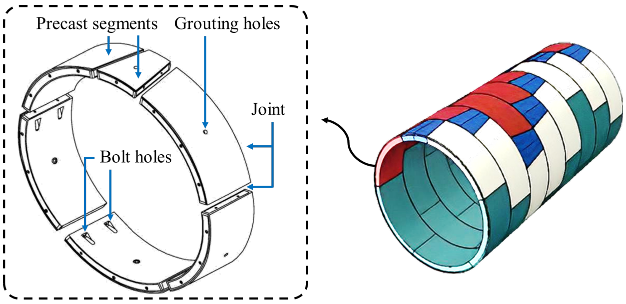



Figure 1. Shield tunnel lining structure and its components

In the context of tunnel surveys, on-site visual inspection plays an important role. While this method leverages substantial knowledge and extensive experience to achieve reasonably accurate results, it is often time-consuming and inefficient. For instance, an inspector may only cover a few kilometers per day (Li et al., 2021, Zhang et al., 2022). Additionally, this approach is generally conducted in harsh environments and requires significant physical effort, making it unsuitable for large-scale tunnel surveys. The presence of active metro operations within tunnels further compromises inspector safety, and prolonged metro shutdowns for survey purposes can disrupt regional daily life. These challenges in tunnel construction and maintenance are common across the globe.

To address these limitations, recent research efforts have explored the utility of surveying methods including total stations, terrestrial laser scanning, as well as specialized mobile laser scanning equipment to acquire tunnel point clouds. Compared to traditional on-site visual surveys and image-based methods, point clouds offer several advantages, such as precise geometric representation of target objects, rapid data acquisition, and lower operational costs. Furthermore, they enable accurate reconstruction and detection of objects through detailed descriptions of 3D geometry and additional point attributes like intensity, RGB color, and normal vectors (Zhang et al., 2019). Consequently, 3D point clouds are highly suitable for use as the primary data in tunnel inspection. However, the complex tunnel environment, with various objects supporting tunnel construction or metro operation, such as power cables, tracks and pipes, often produces numerous interfering artefacts in the point cloud, which can adversely affect the inspection of tunnel lining structures (Lin et al., 2024, Ji et al., 2023, Soilán et al., 2020, Li et al., 2023). Therefore, accurate semantic segmentation of complex tunnel point clouds is very important for checking the assembly quality, deformation, or defects. While deep learning approaches to semantic segmentation of point clouds have achieved remarkable success in various applications, the application of these methods to point clouds of shield tunnels is challenged by the scarcity of training examples for the various objects present in the data.

Building upon recent advancements in point cloud segmentation, we propose a training-free non-parametric network architecture, Tunnel-NN, specifically designed for semantic segmentation of shield tunnel point clouds. To evaluate the performance of Tunnel-NN, its segmentation results are compared with those of PointNet and PointNet++ using a dataset of five point clouds of shield tunnel rings. The results demonstrate that Tunnel-NN achieves an accuracy comparable to the aforementioned learning-based methods. In addition, to enhance the applicability of the network model to large-scale point clouds, such as a single-ring of a tunnel containing millions of points, we propose a sector splitter sampling method based on the geometric characteristics of the tunnel. This approach preserves the local geometric features of the point cloud while reducing the input scale, thereby achieving a balance between computational resource consumption and efficiency. The test results demonstrate that the sector splitter sampling not only significantly enhances the segmentation performance of Tunnel-NN but also yields comparable improvements on PointNet and PointNet++. Compared to widely used machine learning algorithms, tunnel-NN demonstrates superior performance on small sample datasets without the need for training, highlighting its potential for broad applicability in engineering.

The structure of this paper is as follows: Section 2 provides an overview of existing semantic segmentation methods for shield tunnel point clouds. In Section 3, we present a detailed explanation of our proposed framework, Tunnel-NN, along with the sector splitter sampling method. Section 4 reports the experimental results on a dataset we collected. Finally, Section 5 discusses the findings and outlines potential directions for future research.

## 2. Related Work

### 2.1 Semantic Segmentation of Tunnel Point Cloud

To date, segmentation methods for tunnel point clouds can be broadly classified into two main categories.

The first category involves the direct extraction of features from tunnel point clouds using rigorous mathematical theories or fundamental machine learning algorithms. These features are subsequently utilized for clustering, classification, or segmentation to isolate the point clouds associated with the lining structures.

For instance, early research primarily focused on fundamental point cloud features, such as point density, relative distances, and angles, to identify joints and segments in tunnel linings (Xu et al., 2019, Cheng et al., 2019). Building upon this foundation, numerous studies introduced clustering and segmentation techniques on tunnel point clouds using foundational point cloud feature analysis algorithms, including density-based spatial clustering of applications with noise (DBSCAN) (Li et al., 2022), principal component analysis (PCA) (Lamas et al., 2021), and k-nearest neighbors (KNN) (Wang et al., 2022). The cross-sectional design of most shield tunnels is typically circular. As a result, during the construction and operation stages, circular or cylindrical fitting methods can be employed to efficiently extract the lining surface point cloud while filtering out irrelevant internal point clouds, such as those from cables or pipes (Yi et al., 2020, Zhao et al., 2024, Duan et al., 2021). Furthermore, some studies have accounted for the inevitable deformation that occurs during tunnel construction or service life by utilizing elliptical or elliptical cylindrical fitting techniques, which more accurately reflect the actual cross-sectional shape (Xie and Lu, 2017, Zhang et al., 2024a, Wang et al., 2017).

Although these algorithms have demonstrated remarkable performance in relatively simple conditions, their effectiveness tends to diminish when applied to more complex and unstructured tunnel scenarios, where the variability of point cloud quality and noise presents significant challenges. To address these limitations, recent studies have integrated deep learning into tunnel point cloud segmentation, with the goal of improving the accuracy and effectiveness of semantic segmentation in large-scale and complex scenarios, thus forming a second category of methods.

Given the maturity of image semantic segmentation techniques, some studies have proposed transforming the collected 3D tunnel point cloud data into a 2D representation by unfolding the cylindrical surface, creating a 2D image (Cui et al., 2024b, Zhang et al., 2023b). This approach allows the use of well-established models, such as U-Net (Cui et al., 2024a) and Segment Anything Model(SAM) (Kang et al., 2024) for semantic segmentation of the point cloud. By leveraging these advanced image-based techniques, researchers aim to achieve efficient and accurate segmentation of tunnel lining point clouds, bridging the gap between 3D data processing and mature 2D segmentation technologies (Li et al., 2024b, Duan et al., 2021).

Since 2D images compress depth information, these methods may face limitations in accurately capturing the edges of 3D objects and handling overlapping point clouds. Additionally, the complexity of point cloud projection and reprojection can impact the overall efficiency of semantic segmentation. Consequently, deep learning models that operate directly on 3D point clouds continue to hold an indispensable role in achieving more precise and efficient segmentation outcomes (Zhang et al., 2022). In addition to applying advanced 3D point cloud segmentation models like PointNet and PointNet++ (Lin et al., 2024, Soilán et al., 2020), several studies have developed more powerful networks by incorporating techniques such as attention mechanisms (Zhou et al., 2023) and local feature enhancement (Li et al., 2023), further improving segmentation performance.

However, in engineering practice, the high computational cost of training complex semantic segmentation networks, along with the labour-intensive process of point cloud data labeling required for training, remain key barriers to widespread adoption of deep learning methods for semantic segmentation of shield tunnel point clouds (Ji et al., 2023, Ji et al., 2022, Lamas et al., 2021). These challenges motivate our proposal for a training-free semantic segmentation network.

### 2.2 Training-free Network for Point Clouds

As the number of learnable parameters in a typical segmentation model increases, the computational resources required for training and inference grow accordingly. The concept of a non-parametric network for 3D point clouds was first introduced with PointNN (Zhang et al., 2023a). This approach explores a different perspective on 3D point cloud processing, aiming to achieve efficient classification by designing a network that operates without learnable parameters. PointNN classifies a query point cloud by measuring the similarity between the encodings of the query point cloud and those of one or more support point clouds processed through the same network structure. Recently, the application of these networks has been expanded to point cloud semantic segmentation (Zhu et al., 2024).

Building on this approach, researchers have extended the application of non-parametric networksto a variety of point cloud recognition tasks, including pole-like object detection (Zhang et al., 2024b), driving scene perception for autonomous driving systems (Gu et al., 2024), and other applications (Bao et al., 2024, Cheng et al., 2024). This demonstrates the potential of training-free networks for broader use in complex 3D point cloud data analysis.

## 3. Methodology

### 3.1 Tunnel-NN

Our proposed Tunnel-NN is primarily inspired by a study that extended non-parametric networks to the task of point cloud semantic segmentation (Zhu et al., 2024). The architecture of the proposed Tunnel-NN is shown in Figure 2.

As illustrated in Figure 2, the network processes both the query point cloud, which requires semantic segmentation, and the support point cloud, a labeled sample. The detailed processing steps are outlined as follows.

For each point cloud containing $M$ points of $N$ classes, $\{p_i\}_{i=1}^{M}$, input into the network, an initial embedding module is employed to encode the spatial position, $p = (x, y, z)$, and RGB color, $c = (r, g, b)$, of each point, as detailed below:

$$\begin{cases} E(p; u) = [\sin(2\pi up), \cos(2\pi up)] \in \mathbb{R}^{6d}; \\ E(c; u) = [\sin(2\pi uc), \cos(2\pi uc)] \in \mathbb{R}^{6d}, \\ f_0 = E(p; u) + E(c; u) \in \mathbb{R}^{6d}, \end{cases} \tag{1}$$

where $\mathbf{u} = [u_1, \dots, u_d]$ is a list that contains $d$ frequencies, $E(\cdot)$ stands for the embedding, $6d$ represents the combination of 3 coordinates or color channels with 2 functions, $\sin(\cdot)$ and $\cos(\cdot)$. $f_0$ is the output of initial embedding module. Using Eq. (1), the spatial and color information of each point is transformed into a high-dimensional space through a range of different frequency mappings.

After that, in each encoding unit, the input point cloud is first downsampled using farthest point sampling (FPS), with a sampling ratio of 0.5. The point cloud is reduced to half its previous size after each sampling operation. For each discarded point, a KNN search is conducted with the point as the center, and all neighboring points are denoted as $N_p$. Then, for the retained points, which are identified as neighboring points, the features of the center point are concatenated to these neighboring points along the channel dimension as follows:

$$\hat{f}_j^l = \text{Concat}(f^{l-1}, f_j^{l-1}), j \in N_p, \tag{2}$$

where $j$ represents the $j$-th point in the neighborhood $N_p$, and $l$ denotes the $l$-th encoder unit as in Figure 2. In this manner, the characteristics of the discarded points are still preserved. The dimension of $\hat{f}_j^l$ is increased to $2^l \times 6d$ as Eq. (2) is applied at each manipulation layer, with each layer doubling the channel dimension.

Then, for the $j$-th point retained, its feature is manipulated according to the following:

$$f_j^l = W^l \cdot (\hat{f}_j^l + E(\Delta p_j; u) + E(c_j; u))\mathbb{R}^{2^l \times 6d}, \tag{3}$$

where $W^l = [\cos(2\pi)vk] \in \mathbb{R}^{2^l \times 6d}$ is the manipulation weight, $v$ represents a series distributions of sample frequencies, such as Gaussian, uniform, Laplace, and etc, $k = [1, ..., 2^l \times 6d]$, and $\Delta p_j; u$ stands for the position offset between $j$-th point and the center point when applying KNN. By applying Eq. (3), the local features of the point cloud are progressively enhanced.

In the decoding stage, each decoder unit performs two key functions. The first is to restore the downsampled points from the corresponding encoder unit, which means doubling the number of points at each step. The features for these upsampled points are derived through linear interpolation from the points retained in Eq. (2) and Eq. (3). The second function is to concatenate the upsampled features with the features from the point cloud at the same scale, effectively combining information from multiple layers to enhance the final representation.

The output from the final decoder unit has a feature shape of $[M, 90d]$, where $M$ represents the number of points and $90d$ corresponds to the dimensionality of the feature space. This output encapsulates the aggregated and upsampled features from the preceding layers. Finally, the feature arrays of the query point cloud and support point cloud are compared by calculating the similarity to determine the label for each point in the query point cloud, as follows:

$$\begin{cases} \varphi(x) = \exp(-\gamma(1 - x)), \\ \text{logits} = \varphi(F_Q F_P^\top L_P) \in \mathbb{R}^{M \times N}, \end{cases} \tag{4}$$

where $\varphi(x)$ represents the activation function and $\gamma$ denotes the scaling factor. $F_Q$ refers to the output feature from the last decoder unit for query point cloud, while $F_P$ represents the masked average pooling feature of support point cloud, with $F_P$ containing $N$ rows corresponding to $N$ classes. And $L_P \in \mathbb{R}^{N \times N}$ stands for the one-hot labels arrays of $N$ classes.

### 3.2 Sector Splitter Sampling

The proposed sector splitter sampling method is illustrated in Figure 3. For the point cloud of a single ring, the RANSAC cylindrical fitting is applied to extract the cross-sectional center and determine the tunnel axis direction, which is represented as the $z$-axis in Figure 3. As the data in this study is collected using a self-developed simultaneous localization and mapping (SLAM) equipment, the gravity direction ($y$-axis in Figure 3) is not inherently provided. To address this, we start by performing a coarse manual segmentation of the track point cloud. PCA is then applied to this subset of points to determine the principal components, with the direction perpendicular to the plane of the track serving as the estimated $y$-axis, as this direction consistently aligns with gravity.

The $x$-axis direction is determined by calculating the cross product of the $y$-axis and $z$-axis directions. Subsequently, the

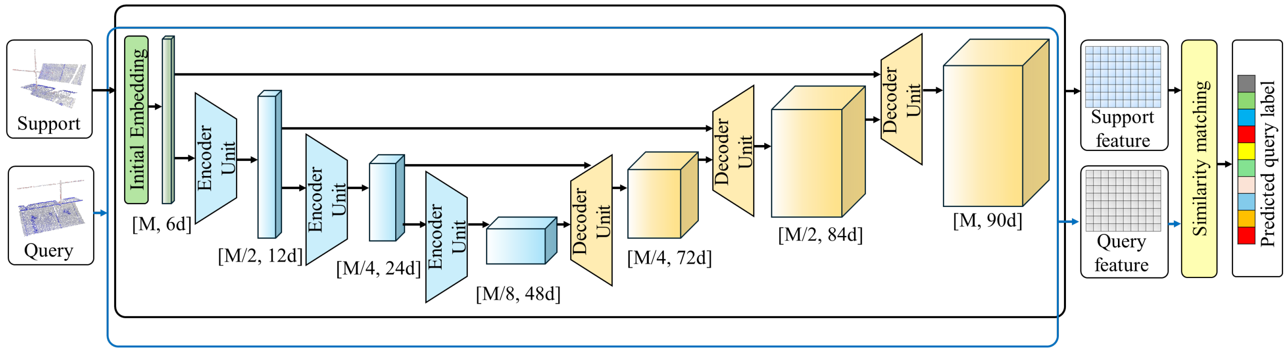



Figure 2. Architecture of Tunnel-NN

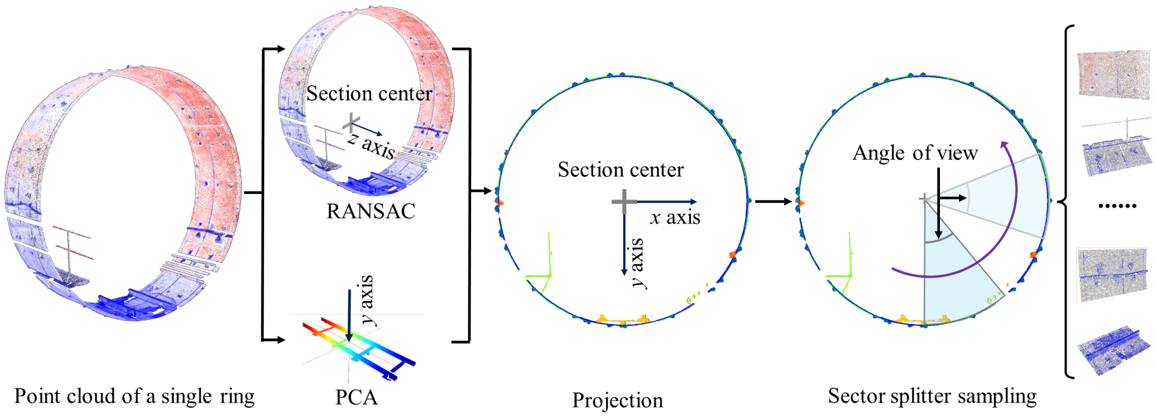



Figure 3. The sector splitter sampling process

point cloud is translated and rotated based on the center coordinates of the cross-section and the $x$, $y$, and $z$ axes, to be projected onto the $x$-$y$ plane. This process ensures that, after projecting each ring of point cloud, the internal components such as the pipes, cables, and tracks are consistently aligned in approximately the same positions across projections.

Next, given a specified angle of view, the first sector is defined with the $y$-axis as the starting reference. All the points within this sector, which are located within the defined angle of view, are extracted to form a distinct partition of the data referred to as a sector. The angle of view is then rotated incrementally without overlapping, ensuring that the entire point cloud is divided into multiple sectors without any omissions. This approach ensures that all sectors maintain the point density of the original point cloud, and have an approximately equal size, thus preserving the original local details.

## 4. Case Study

The data used in this study was collected using a self-developed SLAM equipment in a tunnel under construction in Shenzhen, China. As the original point cloud data only includes coordinate and intensity information, the intensity values were mapped onto a red-white-blue color scale, thereby generating RGB color for the point cloud, as illustrated in Figure 4(a).

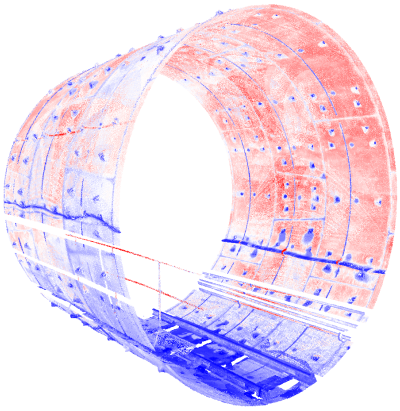
The point cloud data contains a total of eight semantic categories, including segment surfaces, bolt holes, grouting holes, joints, paths, pipes, tracks, and power cables. The annotated point cloud data is displayed in Figure 4(b), with each class represented by a different color. Of these categories, the first four—segment surfaces, bolt holes, grouting holes, and joints—are critical for tunnel deformation analysis.

The annotated point clouds of two rings in the dataset are utilized either as training data for trainable networks or as support data for Tunnel-NN. Point clouds of the other three rings are used to evaluate the performance of the network. The details of the dataset are provided in Table 1. Notably, when using sector splitter sampling, a viewing angle of 10° is applied, with the original 36 samples containing an average of 24,485 points each. Given the input shape limit of 12,288 points, these 36 samples are downsampled by a factor of 0.5 to meet the limit, while doubling the sample count to 72.

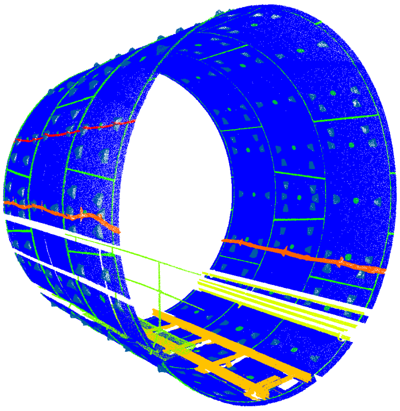
Table 1. Dataset details and specifications

| Type | No. | Number of points | Number of samples | |
|---|---|---|---|---|
| | | | Sector splitter sampling | Random sampling |
| Training or Support | S1 | 878,206 | 72 | 71 |
| | S2 | 970,931 | 72 | 79 |
| Evaluation or Qeury | Q1 | 797,415 | 72 | 65 |
| | Q2 | 825,322 | 72 | 67 |
| | Q3 | 935,465 | 72 | 76 |

To evaluate the performance of Tunnel-NN in comparison with trainable models, two point cloud segmentation networks, namely PointNet and PointNet++, which are commonly used in existing research (Lin et al., 2024, Soilán et al., 2020), are selected as baselines. To assess the effect of sector splitter sampling on model performance, two experimental groups, Group A and Group B, are established. In Group A, all models use the default random downsampling method for dataset preparation, while



(a) RGB point cloud



(b) labeled point cloud

Figure 4. The collected point cloud of shield tunnel

in Group B, the dataset is prepared using the sector splitter sampling method. The input for all network models in both Group A and Group B is limited to 12,288 points, which is the maximum capacity supported by our GPU during the most computationally intensive training phases of our experiments. Training and inference are conducted on an NVIDIA L40 GPU with 48GB of vRAM, and for the trainable models, the number of epochs is set to 500. Three metrics are employed to evaluate model performance: Intersection over Union (IoU) for each class, mean IoU (mIoU) averaged across all classes, and overall accuracy, as detailed below:

$$\mathrm{IoU}_i = \frac{TP_i}{TP_i + FP_i + FN_i}, \tag{5}$$

$$\mathrm{mIoU} = \frac{1}{N}\sum_{i=1}^{N} \mathrm{IoU}_i, \tag{6}$$

$$\mathrm{Accuracy} = \frac{\sum_{i=1}^{N} TP_i}{\sum_{i=1}^{N}(TP_i + FP_i)},, \tag{7}$$

where $TP_i$ represents the number of points that are correctly predicted as class $i$, $FP_i$ denotes the number of points that are incorrectly predicted as class $i$, and $FN_i$ is the number of points that belong to class $i$ but are predicted as other classes. The visualized segmentation results of Group A and B are illustrated in Figure 5 and Figure 6. The quantitative results are detailed in Table 2, with the best results highlighted in bold.

Comparing Figure 5 and Figure 6, we observe that when the models employ random sampling, the segmentation performance for small-sized features such as bolt holes, grouting holes, joints, and power cables is suboptimal. However, using sector splitter sampling significantly enhances the ability of the models to semantically segment these small-sized features. This improvement is further supported by the IoU results in Table 2. For instance, compared to using random sampling, applying sector splitter sampling to Tunnel-NN increases the IoU by 31.9%, 36.2%, 24.6%, and 20.5% for the aforementioned small-sized features, respectively. Additionally, sector splitter sampling yields an improvement of 17.2%, 16.9%, and 16.6% in the mIoU of PointNet, PointNet++, and TunnelNN, respectively.

In the experimental group employing sector splitter sampling, Tunnel-NN achieves an mIoU that surpasses PointNet by 7.6%, though it is 4.5% lower than PointNet++. Tunnel-NN achieved the highest IoU for tracks across all experiments, as well as the highest IoUs for path and pipes in Group B, demonstrating its superior performance in these specific tasks. These results indicate that Tunnel-NN's overall performance is comparable to that of trainable models. An important observation is that while PointNet and PointNet++ required 1.7 hours and 5.2 hours of training time, respectively, Tunnel-NN achieved similar levels of semantic segmentation accuracy without the need for any training. This highlights the efficiency of Tunnel-NN as a non-trainable approach, offering competitive results while significantly reducing computational cost and time.

## 5. Discussion

To compare the point clouds generated by random sampling and sector splitter sampling, the point cloud data obtained from both methods from the same perspective is visualized in Figure 7. In the case of random sampling, the sampled point cloud is uniformly and sparsely distributed across the entire ring, with an average point spacing of 2.647 cm. By contrast, under the same number of samples, the point cloud generated by sector splitter sampling is more densely distributed within a specific sector, resulting in a much finer average point spacing of 0.838 cm.

This difference in point density highlights that sector splitter sampling preserves local geometric details of the tunnel point cloud, such as bolt holes and joints in Figure 7, which are nearly absent in the point cloud obtained through random sampling. Consequently, sector splitter sampling effectively enhances the IoU performance across various semantic segmentation models by retaining these critical features.

In our experiments, a single input of 12,288 points represents the maximum capacity supported by our GPU with 48GB of vRAM during the most computationally intensive training phases, exceeding the capacity of most current desktop GPUs. As the number of input point clouds increases, the additional vRAM required grows proportionally. Therefore, sector splitter sampling can enhance the model's performance when computational resources are limited, which in turn facilitates its broader application to tunnel point cloud datasets.

While Tunnel-NN, when combined with sector splitter sampling, achieves performance comparable to that of PointNet++ without the need for training, there remains potential for further improvement in its semantic segmentation IoU. Future research could explore strategies to enhance its performance, such as optimizing the network architecture or adjusting

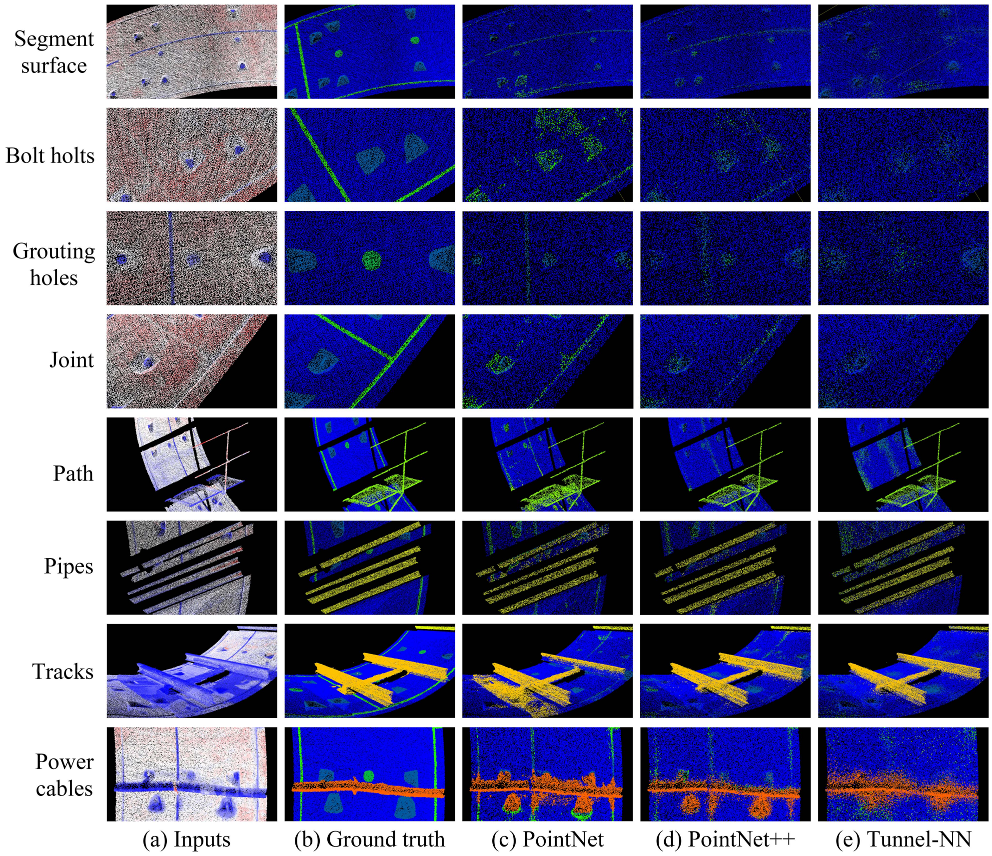



Figure 5. Visualized segmentation results of Group A: models with random sampling

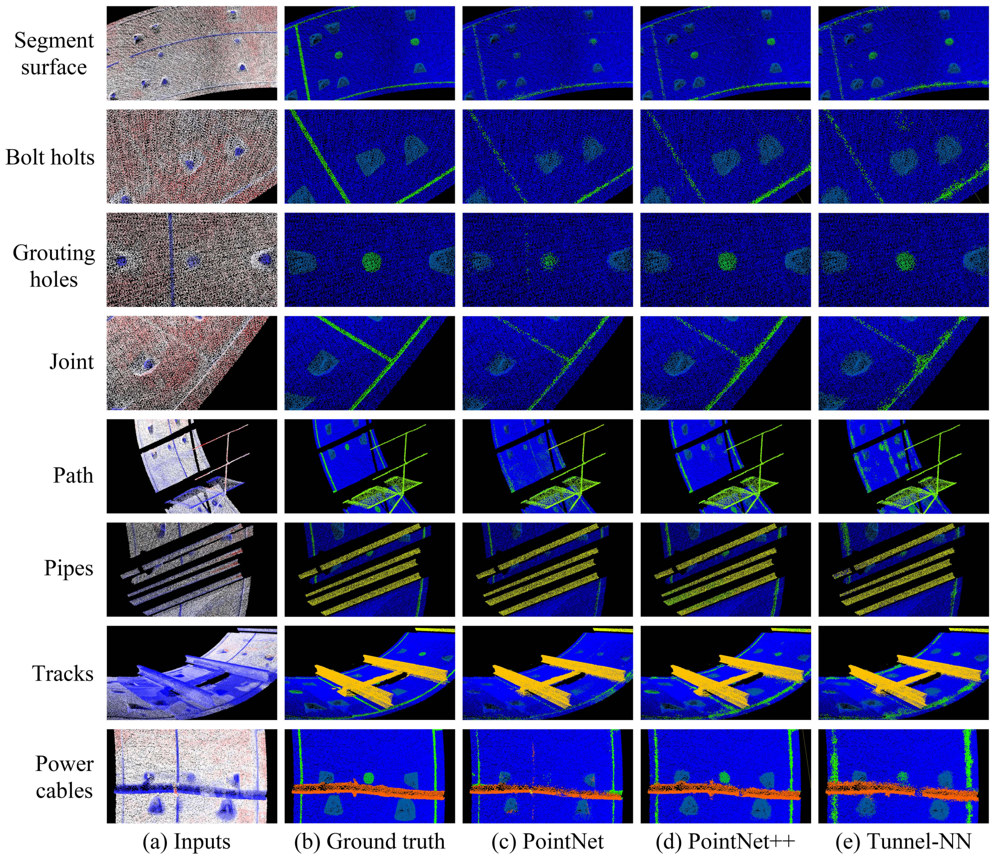



Figure 6. Visualized segmentation results of Group B: models with sector splitter sampling

Table 2. Quantitative performance comparison of different networks (%)

| Group | Method | Training time (h) | IoU per Class | | | | | | | | mIoU | Overall accuracy |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Sgement surface | Bolt holes | Grouting holes | Joint | Path | Pipes | Tracks | Power cables | | |
| A | PointNet | 1.7 | 84.6 | 13.0 | 1.7 | 17.0 | 65.9 | 84.9 | 55.6 | 45.2 | 46.0 | 83.6 |
| | PoinNet++ | 5.2 | 88.0 | 23.0 | 8.1 | 8.4 | **98.7** | **94.9** | 94.7 | 51.3 | 58.4 | 88.2 |
| | Tunnel-NN | 0 | 86.2 | 28.3 | 7.6 | 9.2 | 84.5 | 85.9 | 86.7 | 45.0 | 54.2 | 87.1 |
| B | PointNet | 1.7 | 90.4 | 44.6 | 15.1 | 32.4 | 81.2 | 90.4 | 95.2 | 56.1 | 63.2 | 91.1 |
| | PoinNet++ | 5.2 | **93.9** | **71.9** | **62.5** | **52.1** | 75.9 | 78.7 | 93.3 | **74.2** | **75.3** | **94.3** |
| | Tunnel-NN | 0 | 89.9 | 60.2 | 43.8 | 33.8 | 83.8 | 94.4 | **95.2** | 65.3 | 70.8 | 91.1 |

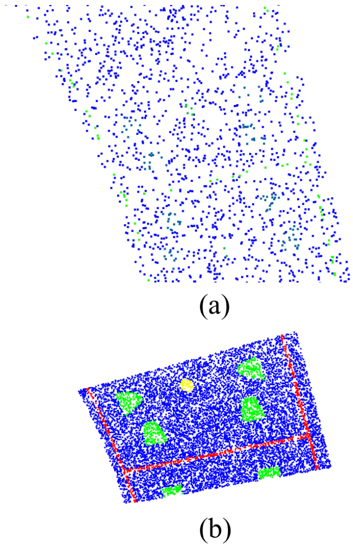



Figure 7. Comparison of point cloud density between random sampling and sector splitter sampling: (a) random sampling, (b) sector splitter sampling

key parameters. Investigating these avenues may provide valuable insights into how Tunnel-NN can be refined to deliver even more accurate segmentation results, making it a promising area for continued study.

## 6. Conclusion

This study presents Tunnel-NN, a training-free semantic segmentation network specifically designed for few-shot shield tunnel point clouds. Our experimental results demonstrate that Tunnel-NN can achieve competitive performance compared to PointNet and PointNet++ on a few-shot dataset while avoiding the need for training. The introduction of the sector splitter sampling method further enhances segmentation accuracy by preserving important local geometric details, such as bolt holes and joints, which are often missed in random sampling methods. Tunnel-NN’s ability to achieve performance comparable to trainable models without the need for training positions it as a promising tool for large-scale engineering applications.

## Acknowledgements

This work was supported by National Key R&D Program of China No.2023YFB3711605, NSFC No.52378405, and Science and Technology Major Project of Department of Transport of Yunnan Province No.202302AD080007.

## References

Attard, L., Debono, C. J., Valentino, G., Di Castro, M., 2018. Vision-based change detection for inspection of tunnel liners. *Automation in Construction*, 91, 142-154.

Bao, Y., Wang, X., Wang, T., Tan, Z., Pu, Y., Chen, K., 2024. Semantic understanding of point clouds based on density-adaptive sampling and position-sensitive feature extraction. *Third International Conference on Electronic Information Engineering, Big Data, and Computer Technology (EIBDCT 2024)*, 13181, SPIE, 262–266.

Cheng, N., Luo, C., Li, X., Hu, R., Li, H., Ma, S., Ren, Z., Jiang, H., Li, X., Lei, S., Li, P., 2024. Rethinking masked-autoencoder-based 3d point cloud pretraining. *2024 IEEE Intelligent Vehicles Symposium (IV)*, 2763–2768.

Cheng, Y.-J., Qiu, W.-G., Duan, D.-Y., 2019. Automatic creation of as-is building information model from single-track railway tunnel point clouds. *Automation in Construction*, 106, 102911.

Cui, H., Li, J., Mao, Q., Hu, Q., Dong, C., Tao, Y., 2024a. STSD:A large-scale benchmark for semantic segmentation of subway tunnel point cloud. *Tunnelling and Underground Space Technology*, 150, 105829.

Cui, H., Mao, Q., Li, J., Hu, Q., Tao, Y., Ma, J., Li, Z., 2024b. 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.-Y., Qiu, W.-G., Cheng, Y.-J., Zheng, Y.-C., Lu, F., 2021. Reconstruction of shield tunnel lining using point cloud. *Automation in Construction*, 130, 103860.

Gu, L., Yan, X., Nan, L., Zhu, D., Chen, H., Wang, W., Wei, M., 2024. PointeNet: A lightweight framework for effective and efficient point cloud analysis. *Computer Aided Geometric Design*, 110, 102311.

Ji, A., Chew, A. W. Z., Xue, X., Zhang, L., 2022. An encoder-decoder deep learning method for multi-class object segmentation from 3D tunnel point clouds. *Automation in Construction*, 137, 104187.

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.

Kang, J., Chen, N., Li, M., Mao, S., Zhang, H., Fan, Y., Liu, H., 2024. A Point Cloud Segmentation Method for Dim and

Cluttered Underground Tunnel Scenes Based on the Segment Anything Model. *Remote Sensing*, 16(1).

Lamas, D., Soilán, M., Grandío, J., Riveiro, B., 2021. Automatic Point Cloud Semantic Segmentation of Complex Railway Environments. *Remote Sensing*, 13(12).

Li, D., Xie, Q., Gong, X., Yu, Z., Xu, J., Sun, Y., Wang, J., 2021. Automatic defect detection of metro tunnel surfaces using a vision-based inspection system. *Advanced Engineering Informatics*, 47, 101206.

Li, J., Zhang, Z., Sun, H., Xie, S., Zou, J., Ji, C., Lu, Y., Ren, X., Wang, L., 2023. GL-Net: Semantic segmentation for point clouds of shield tunnel via global feature learning and local feature discriminative aggregation. *ISPRS Journal of Photogrammetry and Remote Sensing*, 199, 335-349.

Li, T., Li, X., Rui, Y., Ling, J., Zhao, S., Zhu, H., 2024a. Digital twin for intelligent tunnel construction. *Automation in Construction*, 158, 105210.

Li, Y., Shi, J., Xiao, Z., 2022. A Multilevel Point Cloud Classification Method for Underground Tunnels Based on Three-Dimensional Moving LiDAR Measurements. *Mobile Information Systems*, 2022(1), 6181182.

Li, Y., Xiao, Z., Li, J., Shen, T., 2024b. Integrating vision and laser point cloud data for shield tunnel digital twin modeling. *Automation in Construction*, 157, 105180.

Lin, W., Sheil, B., Zhang, P., Zhou, B., Wang, C., Xie, X., 2024. Seg2Tunnel: A hierarchical point cloud dataset and benchmarks for segmentation of segmental tunnel linings. *Tunnelling and Underground Space Technology*, 147, 105735.

Mejia-Dorantes, L., Paez, A., Vassallo, J. M., 2012. Transportation infrastructure impacts on firm location: the effect of a new metro line in the suburbs of Madrid. *Journal of Transport Geography*, 22, 236-250. Special Section on Rail Transit Systems and High Speed Rail.

Peng, J., Peng, F.-L., Yabuki, N., Fukuda, T., 2019. Factors in the development of urban underground space surrounding metro stations: A case study of Osaka, Japan. *Tunnelling and Underground Space Technology*, 91, 103009.

Soilán, M., Nóvoa, A., Sánchez-Rodríguez, A., Riveiro, B., Arias, P., 2020. Semantic segmentation of point clouds with PointNet and KPConv architectures applied to railway tunnels. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, V-2-2020, 281–288.

Wang, J., Zheng, H. B., Huang, H., Ma, G. W., 2017. Point cloud modelling based on the tunnel axis and block estimation for monitoring the Badaling tunnel, China. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-2/W7, 301–306. https://isprs-archives.copernicus.org/articles/XLII-2-W7/301/2017/.

Wang, K., Zhang, Z., Wu, X., Zhang, L., 2022. Multi-class object detection in tunnels from 3D point clouds: An auto-optimized lazy learning approach. *Advanced Engineering Informatics*, 52, 101543.

Xie, X., Lu, X., 2017. Development of a 3D modeling algorithm for tunnel deformation monitoring based on terrestrial laser scanning. *Underground Space*, 2(1), 16-29.

Xu, J., Ding, L., Luo, H., Chen, E. J., Wei, L., 2019. Near real-time circular tunnel shield segment assembly quality inspection using point cloud data: A case study. *Tunnelling and Underground Space Technology*, 91, 102998.

Yi, C., Lu, D., Xie, Q., Xu, J., Wang, J., 2020. Tunnel Deformation Inspection via Global Spatial Axis Extraction from 3D Raw Point Cloud. *Sensors*, 20(23).

Zhang, J., Zhao, X., Chen, Z., Lu, Z., 2019. A Review of Deep Learning-Based Semantic Segmentation for Point Cloud. *IEEE Access*, 7, 179118-179133.

Zhang, R., Wang, L., Wang, Y., Gao, P., Li, H., Shi, J., 2023a. Starting from non-parametric networks for 3d point cloud analysis. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 5344–5353.

Zhang, Y., Ren, X., Zhang, J., Ma, Z., 2024a. A Method for Deformation Detection and Reconstruction of Shield Tunnel Based on Point Cloud. *Journal of Construction Engineering and Management*, 150(3), 04024006.

Zhang, Z., Ji, A., Wang, K., Zhang, L., 2022. UnrollingNet: An attention-based deep learning approach for the segmentation of large-scale point clouds of tunnels. *Automation in Construction*, 142, 104456.

Zhang, Z., Ji, A., Zhang, L., Xu, Y., Zhou, Q., 2023b. Deep learning for large-scale point cloud segmentation in tunnels considering causal inference. *Automation in Construction*, 152, 104915.

Zhang, Z., Khoshelham, K., Shojaei, D., 2024b. Pole-NN: Few-Shot Classification of Pole-Like Objects in Lidar Point Clouds. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, X-4/W5-2024, 333–340.

Zhao, Y., Chen, X., Hu, B., Huang, L., Lu, G., Yao, H., 2023. Automatic monitoring and control of excavation disturbance of an ultra-deep foundation pit extremely adjacent to metro tunnels. *Tunnelling and Underground Space Technology*, 142, 105445.

Zhao, Y., Li, A., Du, Z., Chen, Y., Sun, H., Zhi, Z., 2024. Joint Structure Detection and Multi-Scale Clustering Filtering for Tunnel Lining Extraction From Point Clouds. *IEEE Transactions on Intelligent Transportation Systems*, 25(9), 11214-11226.

Zhou, Y., Ji, A., Zhang, L., Xue, X., 2023. Attention-enhanced sampling point cloud network (ASPCNet) for efficient 3D tunnel semantic segmentation. *Automation in Construction*, 146, 104667.

Zhu, H., Ding, W., Qiao, Y., Xie, D., 2014. Micro-disturbed construction control technology system for shield driven tunnels and its application. *Chinese Journal of Geotechnical Engineering*, 36(11), 1983-1993.

Zhu, X., Zhang, R., He, B., Guo, Z., Liu, J., Xiao, H., Fu, C., Dong, H., Gao, P., 2024. No time to train: Empowering non-parametric networks for few-shot 3d scene segmentation. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3838–3847.