Industrial Engineering and Innovation Management (2025)
Clausius Scientific Press, Canada

DOI: 10.23977/ieim.2025.080118
ISSN 2522-6924 Vol. 8 Num. 1

Tunnel Detection Structure Analysis Based on Neural Network Technology

Yuhu Sun

Aneng Third Bureau Chengdu Engineering Quality Testing Co., Ltd, Chengdu, 611130, China
306787639@qq.com

Keywords: Tunnel Detection; Neural Network; Fault Diagnosis; Health Assessment; Fault Location

Abstract: The existing tunnel detection structural analysis methods have certain problems in fault identification, health assessment and positioning accuracy. In particular, when processing complex tunnel structure data, traditional methods are difficult to adapt to diverse fault types and real-time data changes. To this end, this paper introduces neural network technology, aiming to solve the shortcomings of traditional methods in tunnel detection through deep learning models and improve the accuracy and efficiency of fault detection. This paper adopts a recurrent neural network (RNN) model and combines various sensor data from the tunnel for structural analysis. In the model design, the input layer receives various sensor data such as temperature, pressure, crack width, settlement displacement, etc. from the sensor, and outputs the tunnel health assessment results after multi-layer neural network processing. The model uses the ReLU activation function to optimize nonlinear feature extraction, adjusts network weights through the back-propagation algorithm, and accurately identifies fault types such as cracks, settlement, and deformation. In fault location, through the relationship between network learning and sensor data, the model can locate the fault location and output accurate location results. The experimental results show that in the diagnosis of "crack" faults, the accuracy of the neural network is 90%, much higher than the 75% based on the threshold algorithm; in the diagnosis of "settlement" and "deformation" faults, the accuracy of the neural network is also about 7% and 12% higher than the threshold algorithm. In addition, the accuracy of fault location is also significantly improved. The model can locate faults within an error range of ±1 meter and has a high spatial resolution.

1. Introduction

Deep learning technology has advanced quickly in recent years, opening up new avenues for defect identification and tunnel structure health evaluation. With its powerful feature extraction and nonlinear mapping capabilities, the neural network model can learn the complex patterns of tunnel faults from the data collected by sensors, improving the accuracy and reliability of detection. In view of the characteristics of tunnel structure detection, this paper constructs an intelligent analysis model based on neural network to achieve accurate assessment of tunnel health status, efficient identification of fault types and precise positioning of fault locations.

The main research contents of this paper include: (1) designing a neural network structure suitable for tunnel detection tasks and optimizing the configuration of the input layer, hidden layer and output layer; (2) constructing a health scoring model based on sensor data to improve the quantitative evaluation ability of tunnel stability; (3) combining experimental data to analyze the detection accuracy, fault location accuracy and comparison results of the model with traditional methods. The experimental results show that the proposed neural network model has significant advantages in tunnel health assessment and fault diagnosis, providing an efficient and intelligent solution for tunnel safety monitoring.

This paper first introduces the background and challenges of tunnel health assessment and fault diagnosis, and then proposes a neural network-based intelligent detection model, which focuses on solving the shortcomings of traditional detection methods in terms of accuracy and efficiency. Then, the design of the model is described in detail, including the processing of input data, the optimization of network structure, and the identification and location methods of fault types. The experimental part verifies the significant advantages of the neural network model in fault identification and location accuracy by comparing with traditional methods. Finally, the paper analyzes the experimental results, discusses the limitations of the model and future development directions, and looks forward to the potential of this technology in practical applications.

2. Related Work

Tunnel inspection and structural health monitoring are key issues to ensure safe tunnel operation. With the continuous development of technology, more and more innovative methods have been proposed to improve inspection efficiency and accuracy. The following are some of the progress and research results achieved in the field of tunnel inspection in recent years. Sung et al. installed longitudinal tracks on the top of the tunnel to obtain high-quality image data of the entire length of the tunnel while moving the tracks. In the experiment, image data of a 1,000-meter-long and 20-meter-wide tunnel were obtained, with an overlap rate ranging from 0.02% to 8.36% [1]. Xue and Kale proposed a new method for tunnel inspection that combines a deep learning model and a three-dimensional reconstruction method based on motion reconstruction (SfM-Deep learning). This method visualizes the leakage defects and quantifies the leakage area of the tunnel by projecting the image texture after defect identification onto the three-dimensional model [2]. To perform automated underground inspection, Shim et al. created a management system and inspector robot. The robot has a ground vehicle, a manipulation arm, and computer-controlled sensors. It can use imaging technology to identify damage occurring on the concrete layer and drive itself through the recesses [3]. Lei et al. presented the improved Mask R-CNN, a deep learning-based categorization and delineation system that combines a deformable convolutional network with the sophisticated Matrix attention mechanism. According to experimental data, the model has great stability and efficiency and performs better in terms of detection accuracy than other methods. Additionally, a smartphone app for inspecting the surface of tunnels was created [4]. When Rosso et al. juxtaposed the ResNet-50, EfficientNet-B0, and Vision Transformer (ViT) models for classification of ground-penetration radar photographs in road a tunnel lining assessment, they discovered that the classification algorithm without Fourier coefficient preprocessing outperformed the others [5]. Antoniou et al. used a two-dimensional plane strain finite element analysis model to consider the dynamic interaction between soil and tunnel and simulate the degradation of strength and stiffness of steel reinforcement and concrete caused by corrosion. The study showed that although low-amplitude seismic events did not cause significant structural response, corrosion degradation had a significant impact on the seismic performance of the tunnel [6]. Tumrate et al. discussed in detail the data collection, processing technology, segmentation methods and limitations

of AI technology in structural health monitoring, and looked forward to the future research direction of life-critical infrastructure monitoring [7]. Negi's research showed that the combination of IoT, AI and ML can enhance infrastructure monitoring and reduce human intervention [8]. Elbaz et al. proposed a shield tunnel path prediction optimization model based on reinforcement learning. Through the Q-Table, the model explores and makes use of the shield machine's implicit knowledge by combining the heuristic gray wolf algorithm with the Q-learning has been networking. An realistic tunnel construction example in Tianjin was used to test the model. The results demonstrated that the framework's accuracy for prediction was high when compared to the deep learning approach [9]. Mahmoudi and Rajabi used FLAC3D finite difference software to study the effect of castor holes on tunnel behavior and deformation. Taking the Shiraz-Bushehr railway tunnel as an example, the study found that when the size of the castor hole increases, the axial force of the tunnel increases, which is manifested in different changes when the hole is located below, on the top or on the side of the tunnel [10]. Bitra and Komanapalli focused on Tunnel Field Effect Transistor (TFET) devices and explored their application in next-generation biosensors, especially for instant detection needs in real-time applications. Compared with traditional FET biosensors, biosensors based on junction-free and doped-free TFETs overcome the performance limitations of short channel effects [11]. Although existing tunnel detection methods have made significant progress in improving detection accuracy and efficiency, they still face bottlenecks such as high equipment cost, complex data processing and limited detection range.

3. Method

3.1 Neural Network Model Design

Recurrent neural networks (RNNs) are suitable for processing tasks with time series data. In tunnel health monitoring, they can especially capture the temporal dependencies of sensor data such as stress and vibration. Through the recursive structure, RNNs can remember the state of the previous time step and affect the output at the current time step.

In the RNN model, the state of the hidden layer is updated by the following formula:

$$h_t = \sigma(W_h h_{t-1} + W_x x_t + b_h) \quad (1)$$

Among them, h_t is the hidden layer state at time step t; h_{t-1} is the hidden layer state at the previous time step; x_t is the input data (sensor data) at the current time step; W_h and W_x are the weight matrices of the hidden layer and input layer, respectively; b_h is the bias term; $\sigma(.)$ is the activation function.

3.2 Input Layer Design: Multi-dimensional Feature Input of Sensor Data

Tunnel health monitoring tasks usually need to process multi-dimensional data from multiple sensors. In order for the neural network to better understand the information provided by different sensors, these multi-dimensional data need to be fused into a suitable format and input into the RNN. Assuming that we have multiple sensors collecting data at different time points, and the data of each sensor can be regarded as a feature vector. The input data can be represented as a sequence:

$$\mathrm{X} = [\mathrm{x}_1, \mathrm{x}_2, \cdots, \mathrm{x}_\mathrm{T}] \quad (2)$$

X is an input sequence with a time step of T, and x_t represents the sensor data at time t (multi-dimensional data, including strain, vibration, etc.).

The design goal of the input layer is to preprocess and normalize the data according to the characteristics of different sensors, so that the data of each sensor can be processed by the neural

network at the same scale. The preprocessing formula is as follows:

$$x_t^{'} = \frac{x_t - \mu}{\sigma} (3)$$

Among them, x_t is the sensor data; μ and σ are the mean and standard deviation of the sensor data, respectively; $x_t^{'}$ is the normalized data.

3.3 Hidden Layer Design:

In RNN, the hidden layer structure is its core. In order to enhance the expressive power of the model, we introduced multiple hidden layers in the hidden layer and used LSTM to prevent the gradient vanishing problem. By adjusting the gating mechanism in LSTM, it is possible to effectively memorize important historical information while forgetting irrelevant parts, thereby increasing the sensitivity to changes in the health status of the tunnel structure.

3.4 Output Layer Design:

In tunnel structural health monitoring, the design of the output layer usually depends on the specific task objectives, such as crack identification, structural health assessment, etc. The output layer can be designed as a classification or regression task:

Structural health assessment: A continuous value can be output through the regression model to represent the health score of the tunnel.

Crack detection: If the task is crack identification, the output can be obtained through the classification layer to distinguish whether cracks exist or not.

The formula of the output layer can be expressed as:

$$y_t = W_y h_t + b_y (4)$$

y_t is the model prediction value (such as health score or crack label) at time step t; h_t is the last hidden state of the RNN; W_y and b_y are the resulting layer's biases and weights are altered accordingly.

For crack detection, the output layer can use a softmax activation function to classify whether there is a crack:

$$\hat{y}_t = softmax(W_y h_t + b_y) (5)$$

Among them, $\hat{y}_t$ represents the predicted probability of cracks.

3.5 Structural Analysis and Fault Diagnosis

In tunnel health monitoring, structural analysis and fault diagnosis are one of the core tasks. Through the neural network-based model, the tunnel health assessment can be effectively performed, potential faults can be identified, and detailed fault location information can be provided. This process can be divided into two main parts: tunnel health assessment and fault location and diagnosis.

Tunnel health assessment is to analyze the collected time series data through a neural network model to score the overall stability and safety of the tunnel structure. The purpose of health assessment is to give a quantitative score to reflect the passageway's present state regarding wellness, which is convenient for subsequent maintenance and repair work. This process mainly relies on the neural network to learn from the data of various sensors (such as vibration, stress, displacement, etc.) and then evaluate the health status of the tunnel.

Assuming that the output of the neural network model is a structural health score S which represents the stability of the tunnel structure and has a value range of $[0,1]$, where 0 represents complete instability and 1 represents complete stability. The model is evaluated using the following formula:

$$S = f(h_t) (6)$$

Among them, S is the output health score; $f(.)$ is the mapping function learned by the neural network model; and h_t is the hidden state obtained from the tunnel sensor data.

To achieve this health assessment, the LSTM model processes the time series data $X = [x_1, x_2, \cdots, x_t]$ from different sensors and provides the input data x_t of each time step to the network. By updating the hidden state, the health score is finally obtained.

By training the model, the network can adaptively adjust the weights according to the training data, and then output a score that matches the health status of the tunnel. The loss function of the model can be designed as the mean square error (MSE) to minimize the difference between the predicted health score and the actual score:

$$L = \frac{1}{T}\sum_{t=1}^{T}(S_t - \hat{S}_t)^2 (7)$$

Among them, S_t is the actual health score; $\hat{S}_t$ is the health score predicted by the neural network model; T is the number of samples.

The goal of fault location and diagnosis is to identify potential structural problems such as cracks, deformation, settlement, etc. based on real-time collected data and provide specific fault locations. Through the neural network, the model can find abnormal signals in multi-dimensional sensor data and give the specific location of the fault through the positioning algorithm.

For fault diagnosis, the goal of the model is to identify and locate problems such as cracks, deformation, and settlement in the tunnel. Supposing we use a classification model to output the type of fault or whether a fault exists, and assign a probability value to each fault.

The output layer provides a probability distribution of various defect kinds and classifies them using the soft maximum operate:

$$P(y = k|x_t) = \frac{\exp(W_k h_k + b_k)}{\sum_{j=1}^{K}\exp(W_j h_t + b_j)} (8)$$

$P(y = k|x_t)$ is the probability that the model predicts fault type k at time step t; W_k and b_k are the weight and bias of the output layer, corresponding to fault category k, respectively; h_t is the hidden layer state corresponding to time step t; K is the total number of fault categories (such as cracks, settlement, deformation, etc.).

Through neural network training, the model can learn the characteristics and patterns of faults based on historical data, thereby identifying and locating faults. The specific fault type can be determined by maximizing the classification probability:

$$\hat{y}_t = \underset{k}{argmax} P(y = k|x_t) (9)$$

$\hat{y}_t$ is the predicted fault type at time step t.

3.6 Fault Location and Spatial Analysis

Fault location is not only the classification of fault types but also the determination of their spatial location in the tunnel. To achieve this, the network needs to process sensor data at various locations in the tunnel and analyze the data characteristics of each location to determine whether there is a fault.

Assuming we have multiple sensors distributed at different locations in the tunnel, each sensor corresponds to a location p_i, and the data of sensor i is represented as $x_{i,t}$. In this case, fault location can be identified by comparing the prediction of each location with the actual location through the network.

Through spatial mapping, the model can output the fault probability of each sensor location:

$$P(\text{fault}_i) = \sigma(W_i h_i + b_i) \quad (10)$$

$P(\text{fault}_i)$ is the probability of failure at position i; h_i is the hidden layer state corresponding to position i; $\sigma(.)$ is the sigmoid activation function, which is used to map the output to the interval [0, 1].

If $P(\text{fault}_i) > \theta$, then position i is considered to have a failure, and θ is the set threshold.

4. Results and Discussion

4.1 Data Collection and Preprocessing

First, multi-dimensional sensor data needs to be collected from the tunnel. This data can come from a variety of sensors, such as equipment for the environment, displacement, tension, accelerometers that measure etc.

This paper deploys multiple sensors (such as strain, displacement, vibration, etc.) at different locations in the tunnel to collect data regularly, ensuring that the data is collected once per second to provide sufficient time resolution. The data includes structural data of the tunnel at different construction stages or external conditions, covering normal and fault states, providing a basis for subsequent health assessment and fault diagnosis.

In the data preprocessing stage, filters (such as low-pass filters) are used to remove noise from the data, and the sensor data is normalized to ensure that all input data are within the same scale range. At the same time, the time series data is divided into time windows of fixed length to fully capture the dynamic changes of the tunnel structure.

4.2 Evaluation and Validation

The experiment uses mean square error (MSE) to measure the difference between the health score predicted by the model and the actual score.

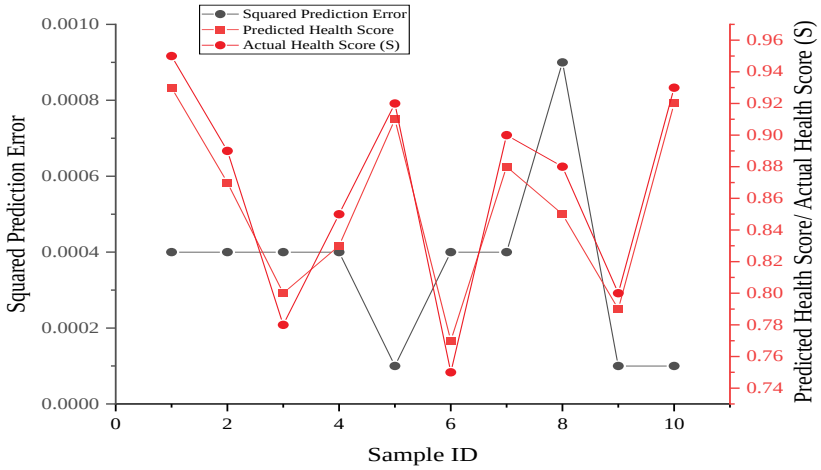

Figure 1. Health score and model predicted health score

The prediction error is the difference between the actual health score and the model predicted

health score, and the square of the prediction error is used to calculate the mean square error (MSE) of the model. As can be seen from Figure 1, the prediction error of most samples is small, indicating that the model can accurately predict the health score of the tunnel. For example, the actual health score of the first sample is 0.95, while the model predicts a health score of 0.93, with a prediction error of 0.02 and a square of the prediction error of 0.0004. The squared values of these errors reflect that the model's performance on each sample is relatively accurate. In particular, in samples 5, 9, and 10, the prediction errors are very small, all within 0.01, indicating that the model's prediction accuracy is high. However, the prediction errors of some samples are slightly biased. For example, the prediction error of the 8th sample is 0.03, and the square of the prediction error is 0.0009, which is slightly larger than the errors of other samples, indicating that the model's prediction on this sample has a certain deviation.

According to the experimental data, Table 1 records the actual situation of different fault categories and the fault categories predicted by the model, as well as the results of whether they match. Through analysis, the classification accuracy and misclassification of the model in fault diagnosis can be evaluated.

Table 1. Classification accuracy

Actual Fault Type	Predicted Fault Type	Match (Correct Prediction)	Error Type
Crack	Crack	Yes	None
Settlement	Settlement	Yes	None
Deformation	Deformation	Yes	None
Crack	Deformation	No	Misclassification
Settlement	Crack	No	Misclassification
Deformation	Deformation	Yes	None
Crack	Crack	Yes	None
Deformation	Deformation	Yes	None
Settlement	Settlement	Yes	None
Deformation	Crack	No	Misclassification

Judging from the experimental results in Table 1, the model performs relatively stably in diagnosing faults such as "cracks", "settlement" and "deformation", and the prediction results completely match the actual fault categories in most cases. For example, for the sample with the fault category of "cracks", the model successfully predicts it as "cracks" and there is no error (no misclassification). Similarly, for the samples with the fault categories of "settlement" and "deformation", the model's predictions are also completely consistent with the actual categories and there is no misclassification. However, in some cases, the model misclassified. For example, in the "crack" fault sample, the model incorrectly predicted it as "deformation", and in the "settlement" fault sample, the model incorrectly predicted it as "crack". Both of these misclassifications indicate that the model may not be able to accurately distinguish similar types of faults in some complex fault scenarios, especially when different fault types have similar sensor data features. Overall, the model has a high fault prediction accuracy and can accurately diagnose the fault category in most cases. Specifically, the correct classification rate of the model is 80%, that is, 8 samples are predicted correctly and 2 samples are misclassified.

According to the experimental data, the error between the actual fault location and the model-predicted fault location is recorded in Figure 2. The location error refers to the difference between the actual fault location and the model-predicted location. This indicator is used to evaluate the fault location accuracy of the model.

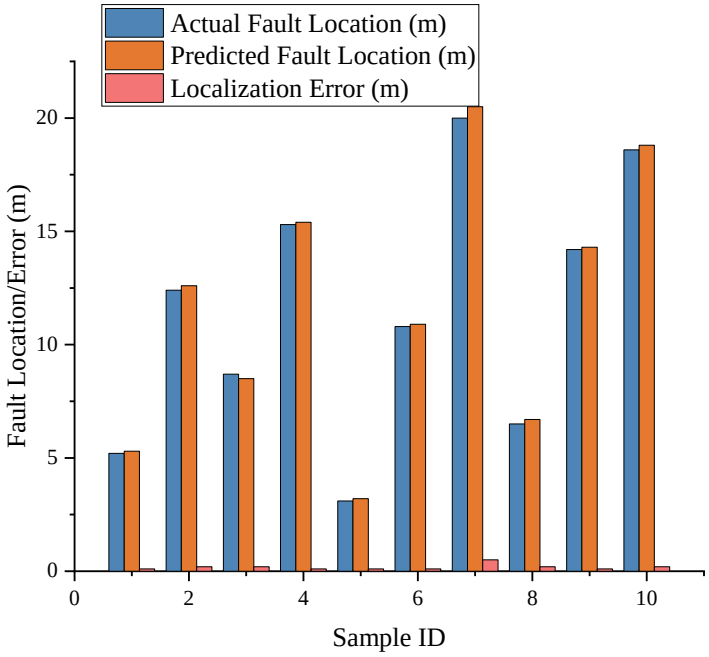

Figure 2. Fault location accuracy

From the fault location data in Figure 2, it can be seen that the actual fault location and the predicted fault location are almost matched, and the location error does not exceed 0.5m. Most errors are only 0.1m. The above data show that the model can accurately predict the fault location in most cases.

According to the experimental results, Figure 3 shows the performance comparison of the neural network model and the threshold-based algorithm in fault type diagnosis, including accuracy, precision, recall, and F1 score. These indicators can be used to comprehensively evaluate the performance of the two algorithms in crack, settlement, and deformation fault diagnosis.

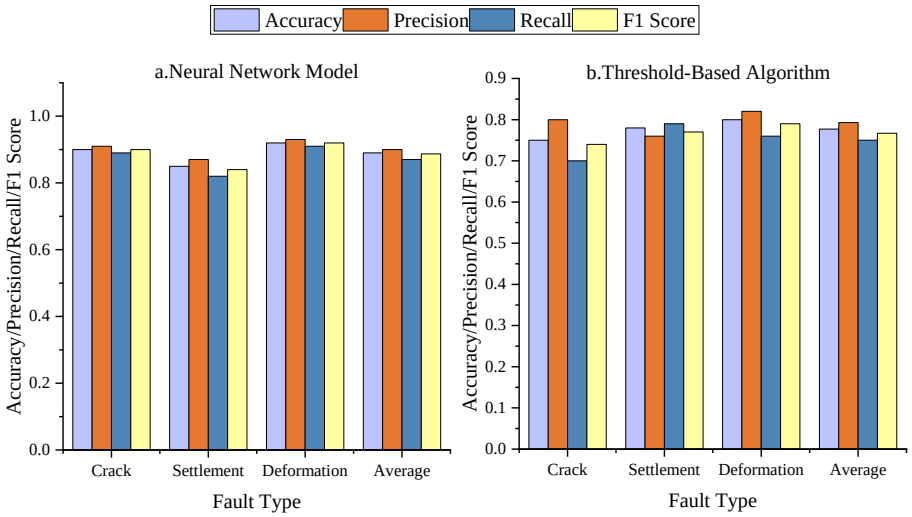

Figure 3. Performance comparison of different models

In terms of accuracy, the neural network model performs significantly better than the threshold-based algorithm on all fault types. For example, in the diagnosis of "crack" faults, the neural network's accuracy is 90%, much higher than the 75% based on the threshold algorithm; in the diagnosis of "settlement" and "deformation" faults, the neural network's accuracy is also about 7% and 12% higher than the threshold-based algorithm, respectively. Overall, the average accuracy of

the neural network model is 89%, compared to 77.7% for the threshold-based algorithm. In terms of accuracy, the neural network model also performs well, especially in the identification of "crack" faults, with an accuracy of 91%, significantly higher than the 80% based on the threshold algorithm. The accuracy rate indicates the proportion of positive samples correctly predicted by the model to all predicted positive samples. The higher accuracy rate indicates that the neural network can effectively avoid misjudgment. Therefore, the neural network performs better in avoiding false positive errors. In terms of recall rate, the neural network model also achieves good performance on all fault types, especially in the recall rate of "deformation" faults, reaching 91%, which is 15% higher than the 76% based on the threshold algorithm (as shown in Figure 3a and Figure 3b). This result shows that the neural network model not only has an advantage in accuracy and precision but also surpasses the traditional threshold-based fault detection method in comprehensive performance.

5. Conclusion

In order to address the issues of low accuracy, low efficiency, and insufficient capacity to recognize complicated fault types in conventional detection methods, this research suggests a neural network-based tunnel health assessment and fault diagnostic method. By designing a neural network model suitable for tunnel detection tasks, the input, hidden and output layer configurations are optimized, effectively improving the accuracy of fault identification, health score assessment and fault location. In the experiment, the model shows high accuracy and precision in different tunnel conditions, especially in fault location, which has obvious advantages over traditional threshold-based algorithms, verifying the application potential of neural network technology in tunnel health monitoring. Through comparative experiments and data analysis, the neural network model in this paper shows high classification accuracy and low positioning error in the identification of common fault types such as tunnel cracks, settlement and deformation, indicating its adaptability and robustness in complex environments. Nevertheless, the training of the model relies on a large amount of labeled data, and there is still room for improvement in performance under certain extreme fault conditions. To lessen the need for labeled data and improve the model's capacity for generalization in various tunnel contexts, future studies can investigate data augmentation and unsupervised learning techniques in greater detail. In addition, the integration of the real-time monitoring system and the adaptive optimization algorithm can further improve the real-time and accuracy of the model, providing a more intelligent solution for tunnel safety monitoring and maintenance.

References

[1] Sung H S, Koh J S. Image-Data-Acquisition and Data-Structuring Methods for Tunnel Structure Safety Inspection[J]. Journal of the Korean Geotechnical Society, 2024, 40(1): 15-28.

[2] Xue Y, Shi P, Jia F, et al. 3D reconstruction and automatic leakage defect quantification of metro tunnel based on SfM-Deep learning method[J]. Underground Space, 2022, 7(3): 311-323.

[3] Shim S, Lee S W, Cho G C, et al. Remote robotic system for 3D measurement of concrete damage in tunnel with ground vehicle and manipulator[J]. Computer-Aided Civil and Infrastructure Engineering, 2023, 38(15): 2180-2201.

[4] Lei M F, Zhang Y B, Deng E, et al. Intelligent recognition of joints and fissures in tunnel faces using an improved mask region-based convolutional neural network algorithm[J]. Computer-Aided Civil and Infrastructure Engineering, 2024, 39(8): 1123-1142.

[5] Rosso M M, Aloisio A, Randazzo V, et al. Comparative deep learning studies for indirect tunnel monitoring with and without Fourier pre-processing[J]. Integrated Computer-Aided Engineering, 2024, 31(2): 213-232.

[6] Antoniou M, Mantakas A, Nikitas N, et al. A numerical case study on the long-term seismic assessment of reinforced concrete tunnels in corrosive environments[J]. Journal of Rock Mechanics and Geotechnical Engineering, 2023, 15(3): 551-572.

[7] Tumrate C S, Saini D K, Gupta P, et al. Evolutionary computation modelling for structural health monitoring of critical infrastructure[J]. Archives of Computational Methods in Engineering, 2023, 30(3): 1479-1493.
[8] Negi P, Singh R, Gehlot A, et al. Specific soft computing strategies for the digitalization of infrastructure and its sustainability: A comprehensive analysis[J]. Archives of Computational Methods in Engineering, 2024, 31(3): 1341-1362.
[9] Elbaz K, Shen S L, Zhou A, et al. Reinforcement learning-based optimizer to improve the steering of shield tunneling machine[J]. Acta Geotechnica, 2024, 19(6): 4167-4187.
[10] Mahmoudi M, Rajabi A M. A numerical simulation using FLAC3D to analyze the impact of concealed karstic caves on the behavior of adjacent tunnels[J]. Natural Hazards, 2023, 117(1): 555-577.
[11] Bitra J, Komanapalli G. A comprehensive performance investigation on junction-less tfet (jl-tfet) based biosensor: Device structure and sensitivity[J]. Transactions on Electrical and Electronic Materials, 2023, 24(5): 365-372.