



OPEN ACCESS

EDITED BY
Manoj Khandelwal,
Federation University Australia, Australia

REVIEWED BY
Kun Fang,
China University of Geosciences Wuhan,
China
Ze Zhou Wang,
National University of Singapore, Singapore

*CORRESPONDENCE
Ning Ma,
✉ maning1989@swjtu.edu.cn

RECEIVED 09 December 2023
ACCEPTED 05 March 2024
PUBLISHED 23 April 2024

CITATION
Ma N and Yao Z (2024), Analysis of slope
stochastic fields using a novel deep learning
model with attention mechanism.
Front. Earth Sci. 12:1352958.
doi: 10.3389/feart.2024.1352958

COPYRIGHT
© 2024 Ma and Yao. This is an open-access
article distributed under the terms of the
[Creative Commons Attribution License \(CC
BY\)](https://creativecommons.org/licenses/by/4.0/). The use, distribution or reproduction in
other forums is permitted, provided the
original author(s) and the copyright owner(s)
are credited and that the original publication
in this journal is cited, in accordance with
accepted academic practice. No use,
distribution or reproduction is permitted
which does not comply with these terms.

Analysis of slope stochastic fields using a novel deep learning model with attention mechanism

Ning Ma* and Zaizhen Yao

Faculty of Geosciences and Environmental Engineering, Southwest Jiaotong University, Chengdu, Sichuan, China

This paper proposes a novel deep learning model incorporating attention mechanisms for the analysis of slope stochastic fields. Initially, a deep learning model is designed to digitally image the stochastic field features of soil strength variability. This is achieved by discretizing the slope soil stochastic field using the Karhunen-Loeve expansion method and transforming the discrete results into digital images. These images are then used to establish a Convolutional Neural Network (CNN) surrogate model that maps the implicit relationship between stochastic field images and slope functional function values, thus calculating the probability of slope failure. The precision of the CNN surrogate model is enhanced through Bayesian optimization and five-fold cross-validation. Moreover, to overcome the limitations of existing data-driven landslide stability prediction models, this study also introduces a Spatial-Temporal Attention (STA) mechanism. By combining the CNN with Long Short-Term Memory (LSTM) networks, the model can accurately approximate the actual results of slope stability calculations in scenarios of high-dimensional representation imaging of stochastic fields and low-probability slope instability. Consequently, this significantly improves the computational efficiency of slope reliability analysis considering stochastic field simulations.

KEYWORDS

spatial variability, slope reliability analysis, convolutional neural network, digital image, attention mechanism

1 Introduction

The physical and mechanical properties of soil are influenced by complex geological processes during formation and environmental changes, leading to spatial variability (Griffiths and Lane, 1999). In geotechnical engineering uncertainty analysis, many studies have highlighted the importance of soil parameter spatial variability for reliability index calculations (Poulos, 1995; Wu and Sidle, 1995; Zhang et al., 2003).

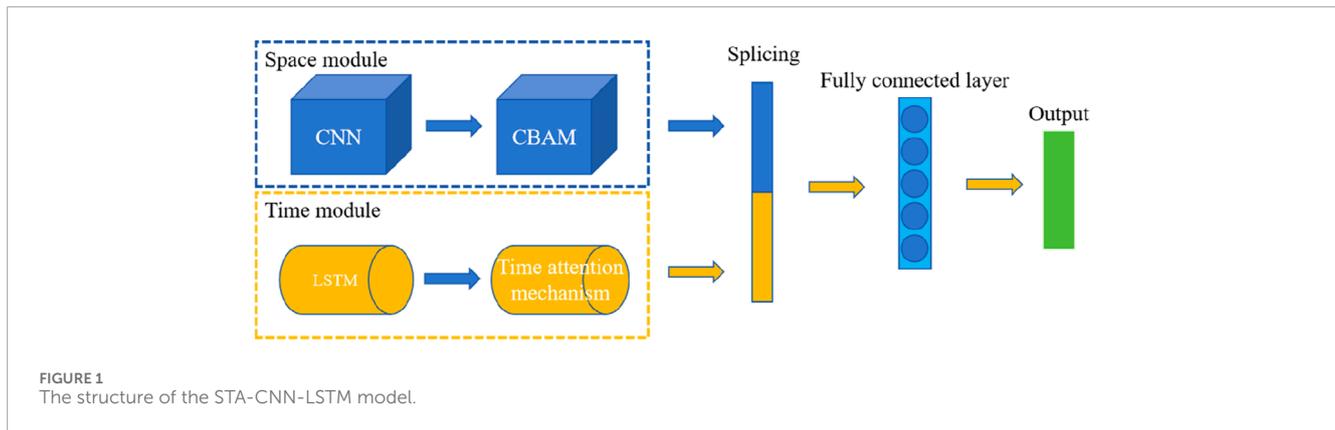
Numerical modeling of spatial variability is crucial in geotechnical engineering. Stochastic field theory offers an accurate framework for characterization in this domain (Sidle, 1992; Malkawi et al., 2000; Zhang et al., 2013). In practical numerical computations, this typically involves discretizing spatial variability into a certain number of random variables, followed by reliability analysis. Although the direct Monte Carlo Simulation (MCS) is favored for its simplicity and ease of implementation, it is computationally intensive and inefficient when dealing with problems of small failure probabilities (Pf) or highly nonlinear implicit function expressions (Liu and Chen, 2007; Lu and Godt, 2008; Zheng et al., 2010). To overcome these challenges, scholars have proposed various

surrogate model methods to approximate these implicit relationships (Fang et al., 2023; Wang et al., 2023). For example, Rahimi et al. (2019) analyzed the Pf of spatially variable soil slopes using an adaptive Kriging method based on error rates. Tan et al. (2020) used a conventional quadratic polynomial response surface for reliability analysis of clay slopes. However, in certain high-dimensional scenarios, traditional surrogate model methods like polynomial response surface and Polynomial Chaos Expansion (PCE) might encounter accuracy and efficiency issues, a problem known as the 'curse of dimensionality' leading to fitting difficulties. Hence, there's a need for more robust surrogate model methods capable of handling high-dimensional situations.

With advances in computing technology, traditional numerical methods are gradually shifting towards intelligent methods, increasingly applied in geotechnical big data and landslide stability prediction. Methods like Artificial Neural Networks (ANN) (Tan et al., 2020), Support Vector Machines (SVM) (Rosenbaum and Rosenbaum, 2003), Extreme Learning Machines (ELM) (Ma and Zhang, 2012), Reciprocal Velocity Method, and Long Short-Term Memory (LSTM) (Shao et al., 2015) neural networks have been instrumental in analyzing complex relationships between landslide displacement and external triggering factors (Ma and Zhang, 2012; Xue, 2016; Chakraborty and Goswami, 2017; Zhang et al., 2021). These intelligent methods, particularly deep learning approaches such as Convolutional Neural Networks (CNNs), have shown great potential in learning the intricate patterns of spatial variability in soil properties and predicting slope stability with high accuracy. Wang and Goh, (2021) introduced a deep learning approach to study slope reliability analysis in spatially variable soils, finding that Convolutional Neural Networks (CNNs) can effectively learn information about random variabilities in both spatial distribution and intensity. The proposed CNN method significantly reduces computational costs while maintaining high prediction accuracy compared to direct random field finite element Monte Carlo simulations. Wang, (2022) introduced a deep learning technique to study geotechnical reliability analysis with multiple uncertainties, finding that Convolutional Neural Networks (CNNs) can effectively handle multiple uncertainties and efficiently predict failure probabilities.

Considering the significant spatial correlation in landslide deformation, particularly in large-scale landslide cases, deformation characteristics at monitoring points vary with spatial location. Recently, some scholars have conducted beneficial research, such as using Segmental Inverse Regression (SIR) for dimensionality reduction of stochastic field expansion coefficients, thereby establishing efficient surrogate models. These methods based on Polynomial Chaos Expansion effectively alleviate high-dimensional problems. Simultaneously, incorporating machine learning methods like Support Vector Machines and Artificial Neural Networks further enhances the accuracy of spatial variability-based analyses of slope stability and foundation bearing capacity. Aminpour et al. (2023) explore the efficiency of machine learning (ML) models and Artificial Neural Networks in predicting slope stability and the probability of failure in spatially variable random fields, with and without factor of safety (FOS) computations. Their study

demonstrates that ML predictions can be highly accurate across a wide range of soil heterogeneity and anisotropy, significantly reducing computational costs by approximately 100 times while maintaining a low error rate in predicted failure probabilities. He et al. (2023) present a framework for ready-to-use deep-learning surrogate models for problems with spatially variable inputs and outputs, introducing three innovations: the creation of surrogate models covering a wide range of material properties and boundary conditions, the handling of large-scale spatially variable data, and the first attempt to use U-Nets as surrogate models for geotechnical problems. Their results demonstrate that while fully connected networks are suitable for simple problems, deep neural networks that account for data structure, such as modified U-Nets, provide better results for complex problems. Sameen et al. (2020) developed a deep learning-based technique for landslide susceptibility assessment in Southern Yangyang Province, South Korea, using a one-dimensional convolutional network (1D-CNN) with Bayesian optimization. The study demonstrated that CNN outperformed traditional models like ANN and SVM, achieving higher accuracy and area under the receiver operating characteristics curve (AUROC) due to its complex architecture and ability to handle spatial correlations. Bayesian optimization further enhanced CNN accuracy by approximately 3%. This approach could be useful in developing landslide susceptibility maps in complex scenarios with non-linear variable contributions. Furthermore, this paper proposes an innovative surrogate model method that transforms the results of stochastic field characterization into digital images and then constructs the STA-CNN-LSTM model using Convolutional Neural Networks (CNN) combined with a Spatial-Temporal Attention (STA) mechanism. This approach establishes a deep learning surrogate model linking stochastic field digital images with functional function values. Bayesian optimization is employed to determine the model's hyperparameters, and five-fold cross-validation is used to enhance the model's accuracy and reliability. The mentioned Spatial-Temporal Attention mechanism is an attention model that integrates spatial and temporal dimensions, aiming to capture the spatiotemporal correlations in landslide deformation data. In our STA-CNN-LSTM model, the STA mechanism is designed as two parallel modules: a Spatial Attention Module (SAM) and a Temporal Attention Module (TAM). The SAM improves the model's ability to capture spatial variability by aggregating spatial features with weighted emphasis on important spatial regions. Specifically, SAM utilizes the output of convolutional layers to generate a spatial attention map, which is normalized by a SoftMax function and then multiplied by the original spatial features to enhance the representation of important regions. The TAM focuses on the temporal dimension by weighting and combining features from different time steps, emphasizing time points with significant impact on slope stability. TAM calculates temporal attention weights using the hidden states of LSTM layers and applies these weights to the time-series features to highlight critical points in reliability analysis. This approach demonstrates the effectiveness and applicability of the STA mechanism in enhancing model performance for reliability analysis of slope stability.



2 Materials and methods

2.1 STA-CNN-LSTM modeling process and model structure

This research proposes a new STA-CNN-LSTM model based on spatial-temporal attention mechanisms for landslide stability analysis (Ding et al., 2020; Chaudhary et al., 2021; Xu et al., 2023). The Convolutional Neural Network (CNN) is utilized to extract correlations in spatial displacement data, and the Long Short-Term Memory (LSTM) neural network is employed to capture the temporal features of external environmental factors. Additionally, a spatial-temporal attention mechanism is introduced to these two deep learning networks. Its role is to reflect the spatial-temporal characteristics of landslide deformation from various aspects and reveal significant triggering factors, thus providing physical interpretability to the predictive model. Specifically, the model is constructed by integrating the Convolutional Neural Network (CNN) with the Long Short-Term Memory (LSTM) neural network. Figure 1 depicts the structure of the proposed landslide deformation prediction model based on spatial-temporal attention mechanisms, the STA-CNN-LSTM. The model, which processes two types of input data (temporal and spatial), consists of two modules: a spatial module and a temporal module. These modules process different types of landslide data in parallel. Spatial Module (CNN) Parameters: Convolutional Layer 1: Contains 32 filters, filter size is 3x3, activation function is ReLU. Pooling Layer 1: Uses max pooling, pooling window size is 2x2. Convolutional Layer 2: Contains 64 filters, filter size is 3x3, activation function is ReLU. Pooling Layer 2: Uses max pooling, pooling window size is 2x2. Fully Connected Layer: Contains 128 neurons, activation function is ReLU. Temporal Module (LSTM) Parameters: LSTM Layer 1: Contains 64 units, activation function is tanh. LSTM Layer 2: Contains 32 units, activation function is tanh.

2.2 Spatial module

(1) CNN

A Convolutional Neural Network is a type of feedforward neural network primarily used for processing data in Euclidean

space (Hakim et al., 2022). When dealing with color digital images (as shown in Figure 2A), data is stored in pixels and represented by values (ranging from 0 to 255) across three channels (red, green, blue). The spatial data input from monitoring points in this paper differs from that of color digital images. For a slope monitoring system (Figure 2B), deformation information is recorded by monitoring points, where the locations of the monitoring points correspond to pixel positions. Different types of deformation data (displacement, velocity, and acceleration) correspond to different channels (Li et al., 2008). Therefore, for the CNN model input, the deformation data from monitoring points will be processed as three-channel images.

Compared to other neural networks, the CNN model includes convolution and pooling operations that extract spatial features by integrating deformation information from different monitoring points. A typical CNN model consists of three layers: the convolution layer, the pooling layer, and the fully connected layer. In the convolution layer, the output feature matrix M is generated by performing convolution operations, which involve moving a filter over the input feature F . Pooling operations usually follow the convolution layer, where adjacent elements of the feature matrix M are integrated using max pooling, average pooling, or other methods. The method of convolution calculation is as Eq. 1:

$$m_{i,j} = \sum_c \sum k \bullet f_{i,j} \quad (1)$$

Where m is an element of the matrix M ; Σ represents the sum of the matrix elements; c is the number of channels in F ; $f_{i,j}$ is a submatrix of F , of the same size as the filter; i and j are the row and column movement steps of the filter k , respectively.

As the filter moves over the input feature, it repeatedly acts on different submatrices $f_{i,j}$ of the input feature map. This is the parameter sharing mechanism of CNN, which reduces the number of parameters and the complexity of the model. Convolution and pooling are specialized mathematical operations on matrix data, giving CNNs an advantage in extracting local spatial features. In training the CNN model, the selection and optimization of hyperparameters are crucial. We adopt the following strategy for hyperparameter optimization:

Learning rate: A learning rate decay strategy is used, with an initial learning rate set at 0.001, decreasing by 10% every five training epochs.

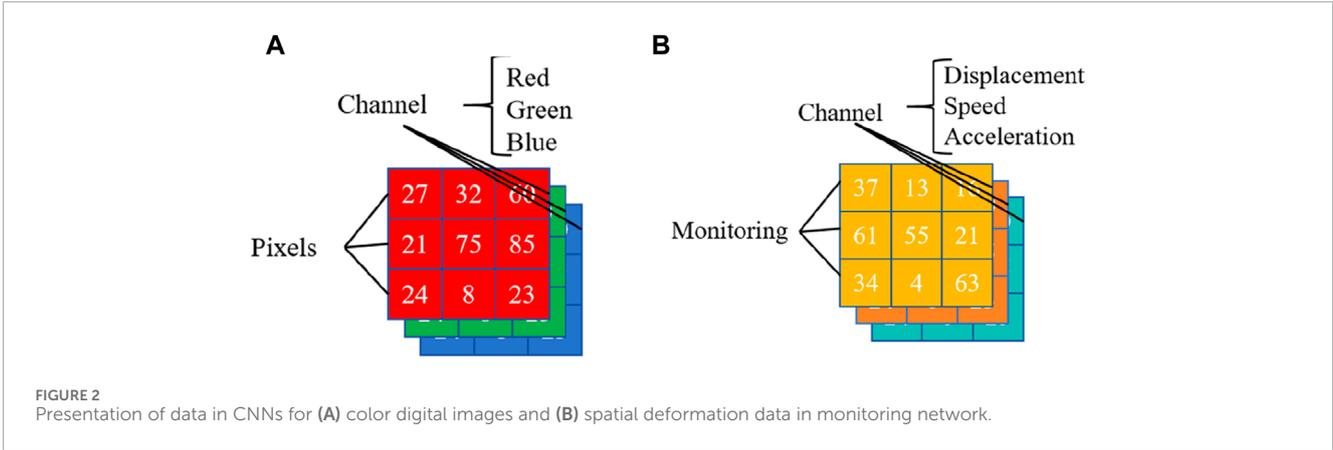


FIGURE 2 Presentation of data in CNNs for (A) color digital images and (B) spatial deformation data in monitoring network.

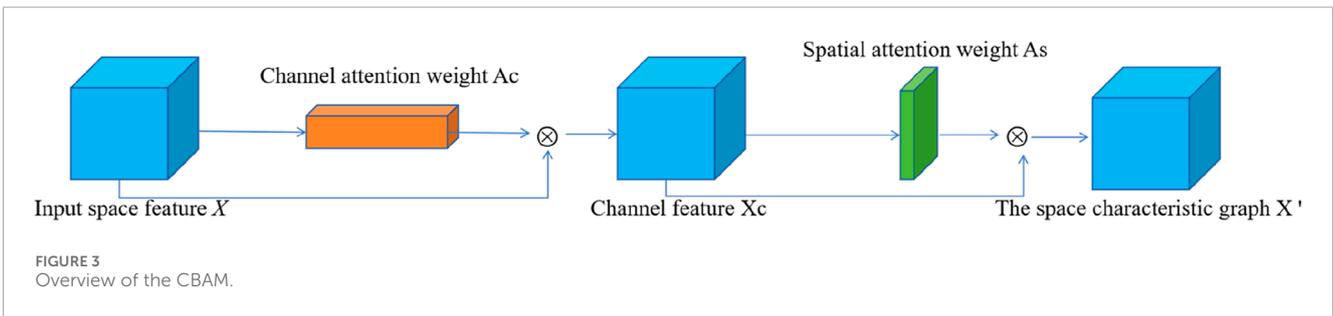


FIGURE 3 Overview of the CBAM.

Batch size: Set to 64, based on available memory and computational resources.

Training epochs: Set between 0 and 1750 epochs, based on the model's performance on the validation set, with early stopping employed to prevent overfitting.

(2) CBAM

CBAM is a lightweight yet effective module that combines channel and spatial attention components. The feature map outputted by a CNN can be represented as the feature X in Figure 3. During the learning process, CBAM determines which channels and location information are more meaningful. For the original feature $X \in \mathbb{R}^{C \times H \times W}$, CBAM successively infers the channel feature map $X_c \in \mathbb{R}^{C \times H \times W}$, and the spatial feature map $X_s \in \mathbb{R}^{C \times H \times W}$ (Figure 3). During the attention process, the size of the original feature does not change, and the detailed feature maps can be represented as Eqs 2, 3.

$$X_c = A_c(X) \otimes X \tag{2}$$

$$X' = A_s(X_c) \otimes X_c \tag{3}$$

Where \otimes represents matrix multiplication, X_c is the result of channel attention, and X' is the feature enhanced by CBAM. A_c is the channel attention weight, and A_s is the spatial attention weight.

Channel attention differentially focuses on different channels of the feature map by increasing the weight of effective channels and reducing the weight of useless channel information. For the original feature map $X \in \mathbb{R}^{C \times H \times W}$, average pooling and max pooling are used

to integrate spatial information within each channel, which can be represented as Eqs 4, 5.

$$A = AvgPool(X) \tag{4}$$

$$M = MaxPool(X) \tag{5}$$

Then, A and M are processed through a shared-weight network composed of a multi-layer perceptron (MLP). The output vector elements are then summed up to obtain the final channel weights, represented by Eq. 6.

$$A_c = \sigma(MLP(AvgPool(X)) + MLP(MaxPool(X))) = \sigma(MLP(A) + MLP(M)) \tag{6}$$

Where A_c is the channel attention weight and σ is the sigmoid activation function. The depth C of $A_c \in \mathbb{R}^{C \times 1 \times 1}$ is the same as the number of channels in the original feature map F , so each position of A_c assigns a weight corresponding to a channel of the original feature map.

For the channel feature map $X_c \in \mathbb{R}^{C \times H \times W}$, average pooling and max pooling operations are applied on the channel dimension. Then, the features after the two pooling operations are integrated and passed through a standard convolution layer. The spatial attention weights can be generated as Eq. 7.

$$A_s = \sigma(f^{7 \times 7}([AvgPool(X):MaxPool(X)])) \tag{7}$$

Where σ is the sigmoid activation function, and f 7×7 is the convolution layer with a kernel size of 7×7 .

2.3 Temporal module

(1) LSTM

Long Short-Term Memory (LSTM) networks are a branch of Recurrent Neural Networks (RNNs). Compared to Artificial Neural Networks (ANNs), RNNs add a state body between hidden layers to store information, enabling RNNs to learn relevant information from time series data inputs. In the structure of LSTM, the state body is replaced by a self-looping memory block, consisting of input gates, forget gates, output gates, and memory cells, providing LSTM with long-term memory and forgetting capabilities. The forget gate determines how much historical information can be stored in the memory cell, the input gate decides how much information can be saved in the memory cell at the current moment, and the output gate's role is to transform the current cell state in LSTM into the next cell state. The gate operations and cell states of the LSTM model can be represented by Eqs 8–13.

$$i_t = \sigma(w_{xv}x_t + w_{hu}h_{t-1} + b_i) \tag{8}$$

$$f_t = \sigma(w_{xy}x_t + w_{hj}h_{t-1} + b_f) \tag{9}$$

$$o_t = \sigma(w_{xo}x_t + w_{ho}h_{t-1} + b_o) \tag{10}$$

$$g_t = \tan h(w_{xc}x_t + w_{hc}h_{t-1} + b_c) \tag{11}$$

$$c_t = f_t \cdot c_{t-1} + i_t \cdot g_t \tag{12}$$

$$h_t = o_t \cdot \tanh(c_t) \tag{13}$$

Where $i_t, f_t, o_t,$ and c_t are the four outputs of the input gate, forget gate, output gate, and storage cell, respectively; $b_i, b_f, b_o,$ and b_c are their corresponding bias vectors. $w_{xv}, w_{xy}, w_{xo}, w_{xc}$ and $w_{hj}, w_{hu}, w_{ho}, w_{hc}$ are the input matrices and hidden matrices of the three gates and storage cell mentioned above. σ is the sigmoid function, and \tanh is the hyperbolic tangent function.

(2) Temporal Attention Module

A temporal attention mechanism is added to the original LSTM to select hidden states throughout the memory cycle. Figure 4 briefly illustrates the process of temporal attention. Initially, the hidden states of LSTM units at different time steps, when presented as input data, are processed through a fully connected neural network, which then outputs temporal attention weights; at each time step, temporal attention weights are allocated to the hidden states. The method of weight distribution by the temporal attention mechanism and its final output are as Eqs 14–16:

$$H = [h_1, h_2, \dots, h_k]_{k \times s} \tag{14}$$

$$\beta = \text{soft max}(A_i(H)) = [\beta_1, \beta_2, \dots, \beta_k]_{1 \times k} \tag{15}$$

$$h^{at} = \sum_{i=1}^k \beta_i h_i \tag{16}$$

Where h_k is the hidden state of the LSTM unit at the k^{th} step, and s is the size of each hidden state. A_i is the fully

connected neural network, and h^{at} is the final output of the LSTM unit.

2.4 Bayesian optimization model

The Bayesian optimization model is a statistical method for global optimization, particularly suitable for optimizing black-box functions with high computational costs or without analytic expressions. It is based on Bayesian theory and Gaussian processes, guiding the search process through a probabilistic model of the objective function, thereby finding the global optimum within fewer iterations. The main steps include:

Prior distribution: Select an appropriate prior distribution (usually a Gaussian process) to describe the uncertainty of the objective function.

Acquisition function: Define an acquisition function to determine the location of the next evaluation point. Common acquisition functions include Expected Improvement (EI) and Probability of Improvement (PI).

Posterior update: After obtaining the observation value of the objective function at the new evaluation point, use the Bayesian rule to update the prior distribution to obtain the posterior distribution.

Iterative optimization: Repeat the above steps until the stopping condition is met (such as reaching a predetermined number of iterations or satisfying the convergence criterion). By fitting a surrogate function to the true objective function and using the acquisition function to automatically select the next optimal sampling point for evaluation, unnecessary sampling is avoided, reducing evaluation costs, and obtaining the optimal solution in fewer iterations. The Bayesian formula used in the hyperparameter optimization process is as Eq. 17.

$$P(f | D_{1:n}) = \frac{P(D_{1:n} | f)P(f)}{P(D_{1:n})} \tag{17}$$

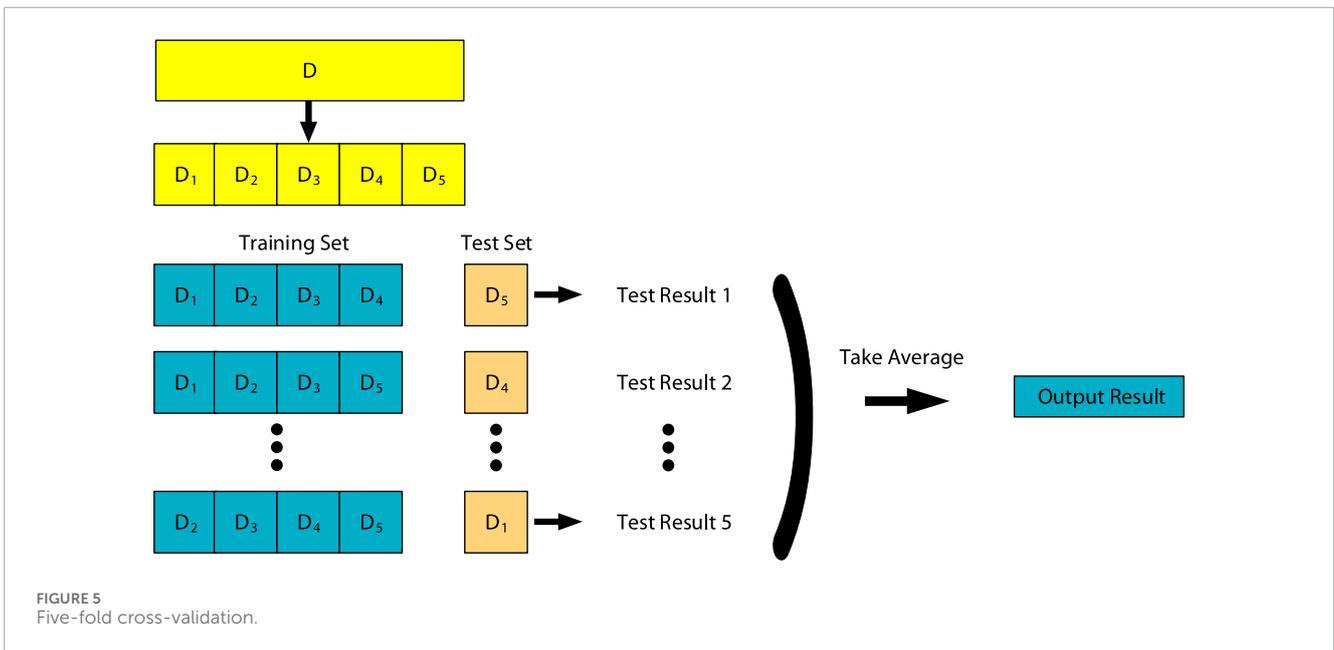
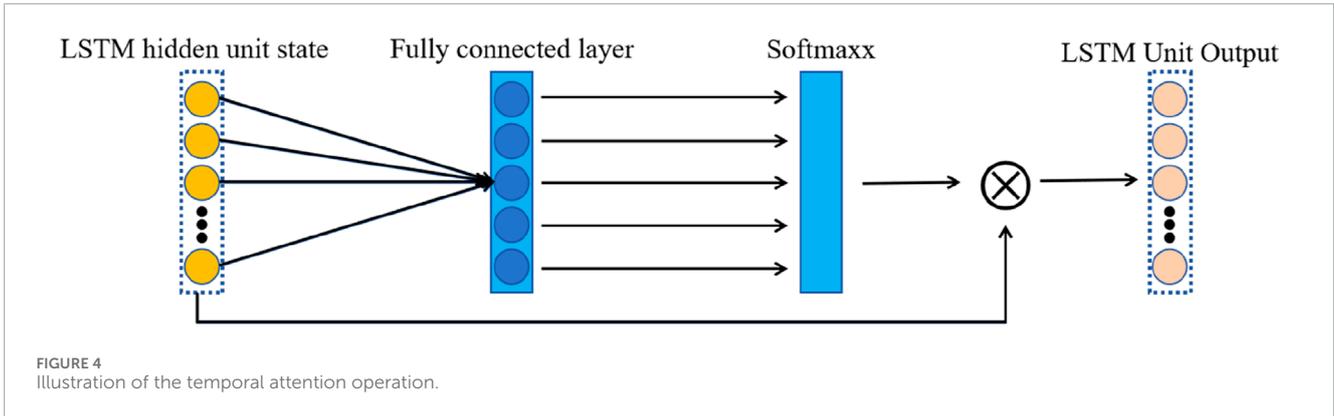
In the formula, $P(f | D_{1:n})$ represents the posterior probability, is the basis for the continuous revision of the unknown objective function f through the observed data set $D_{1:n}$; $P(f)$ represents the prior probability distribution of f ; $P(D_{1:n})$ is the marginal likelihood of $D_{1:n}$; $P(D_{1:n} | f)$ is the likelihood of $D_{1:n}$ given f .

$$\begin{aligned} x^* &= \arg \min_{x \in R} f(x) \\ \text{or} \\ x^* &= \arg \max_{x \in R} f(x) \end{aligned} \tag{18}$$

In the Eq. 18: $f(x)$ represents the objective function of Bayesian optimization. x represents the random variable to be optimized; x^* represents the optimal value obtained after optimization.

2.5 Cross-validation

Cross-validation is a statistical analysis method used to evaluate classifier performance. The basic idea is to divide the original data into groups, with one part serving as the training set and the other part as the validation set. The classifier is first trained on the training set, and then the trained model is tested on the validation



set to evaluate the classifier’s performance. K-Fold Cross-Validation (KCV) divides the original data into K groups, extracting one subset as the validation set without repetition, and combining the remaining K-1 subsets as the training set. This process results in K models, and the average accuracy of these K models on the validation sets is used as the performance indicator of the K-fold cross-validation classifier. K-fold cross-validation can avoid overfitting and underfitting, and the final results are more convincing. As shown in Figure 5, this paper adopts the five-fold cross-validation method.

2.6 Model evaluation metrics

The performance evaluation metrics for the model include Root Mean Square Error (RMSE), Mean Absolute Percentage Error (MAPE), and Coefficient of Determination (R^2), with their respective formulas being:

$$RMSE = \sqrt{\frac{1}{n} \sum_{j=1}^n (Y_f - T_i)^2} \tag{19}$$

$$MAPE = \frac{1}{n} \sum_{j=1}^n \left| \frac{Y_f - T_i}{T_i} \right| \times 100\% \tag{20}$$

$$R^2 = 1 - \frac{\sum_{i=1}^n (Y_i - T_i)^2}{\sum_{i=1}^n (T_i - T_{ave})^2} \tag{21}$$

In these Eqs 19–21, n represents the number of test samples; Y and T are the predicted and actual values, respectively; T_{ave} is the average of the actual values of the test samples. This paper utilizes three metrics - Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and Coefficient of Determination (R^2) - to evaluate the model’s prediction accuracy on the validation and test sets by comparing the predicted values with the actual values.

2.7 Spatial variability of soil parameters

To account for the spatial variability of soil parameters, it is generally necessary to specify the mean, standard deviation, and autocorrelation function of soil properties. Commonly used

autocorrelation functions in the literature include the exponential and Gaussian types:

$$\rho(\tau_h, \tau_v) = \begin{cases} \exp(-\tau_h/L_L - \tau_v/L_v) \\ \exp[-(\tau_h/L_L)^2 - (\tau_v/L_v)^2] \end{cases} \quad (22)$$

In Eq. 22, τ_h and τ_v represent the horizontal and vertical distances between two points, respectively, while L_L and L_v are the horizontal and vertical autocorrelation distances, respectively.

A stationary Gaussian random field assumes that the mean and variance of soil property parameters do not vary with soil layer depth and follow a normal distribution. The expression for the Karhunen-Loève (KL) expansion of such a field as Eq. 23:

$$\widehat{H}(\mathbf{x}) = \mu + \sigma \widehat{H}^D(\mathbf{x}) = \mu + \sigma \sum_{i=1}^M \sqrt{\lambda_i} \varphi_i(\mathbf{x}) \xi_i \quad (23)$$

In the equation, μ and σ respectively are the mean and standard deviation of the normal variable; $\widehat{H}^D(\mathbf{x})$ represents the standard Gaussian random field; \mathbf{x} is the coordinate of the spatial point within the discretized domain of the random field; M is the number of truncation terms; λ_i and $\varphi_i(\mathbf{x})$ are the i^{th} eigenvalue and eigenfunction of the autocorrelation function, respectively; ξ_i is an independent standard normal random variable.

$$\int_{\Omega} \rho(x_1, x_2) \varphi_i(x_2) dx_2 = \lambda_i \varphi_i(x_1) \quad (24)$$

In Eq. 24, Ω represents the discretized domain of the random field, and x_1 and x_2 are the coordinates of any two points within the discretized domain. For the exponential autocorrelation function, the eigenvalues and eigenfunctions have analytic solutions, the specific forms of which can be found in the literature. For the Gaussian autocorrelation function, solutions can be obtained through numerical methods such as the Galerkin method.

Non-Gaussian random fields can be derived from Gaussian random fields through equivalent probability transformations. For soil property parameters that follow a log-normal distribution, the discretization results of the random field as Eq. 25

$$\widehat{H}(\mathbf{x}) = \exp(\mu_{\ln X} + \sigma_{\ln X} \widehat{H}^D(\mathbf{x})) \quad (25)$$

In the formula, $\mu_{\ln X}$ and $\sigma_{\ln X}$ respectively represent the mean and standard deviation after normalization of the log-normal variable. When considering the inter-correlation of multiple random fields, the ξ_i in Eq. 21 needs to be replaced with correlated standard normal random variables ξ_i , $\chi_i \chi_j = \xi_i^T \mathbf{L} \mathbf{L}^T \xi_j = \mathbf{R}_0$ where \mathbf{L} is the upper triangular matrix obtained from the Cholesky decomposition, and \mathbf{R}_0 is the matrix of equivalent inter-correlation coefficients in the standard normal space. For the log-normal random field, the relationship between \mathbf{R}_0 and the inter-correlation coefficient matrix \mathbf{R} in the original space is:

$$R_{0ij} = \frac{\ln(1 + R_{ij} \delta_i \delta_j)}{\sqrt{\ln(1 + \delta_i^2)} \sqrt{\ln(1 + \delta_j^2)}} \quad (26)$$

In Eq. 26, δ represents the coefficient of variation of the soil property parameters.

When discretizing the random field, it is common practice to only retain the first M largest eigenvalues and eigenfunctions from the Karhunen-Loève (KL) expansion to reduce the number of random variables and lower the dimensionality. To ensure the accuracy of the random field discretization, the expected energy ratio factor ε is used to measure the accuracy of the random field discretization. The number of expansion terms M is determined based on the ε threshold value (typically taken as $\varepsilon = 0.95$).

$$\varepsilon = \frac{\sum_{i=1}^M \lambda_i}{\sum_{i=1}^{\infty} \lambda_i} = \frac{\sum_{i=1}^M \lambda_i}{L_h L_v} \quad (27)$$

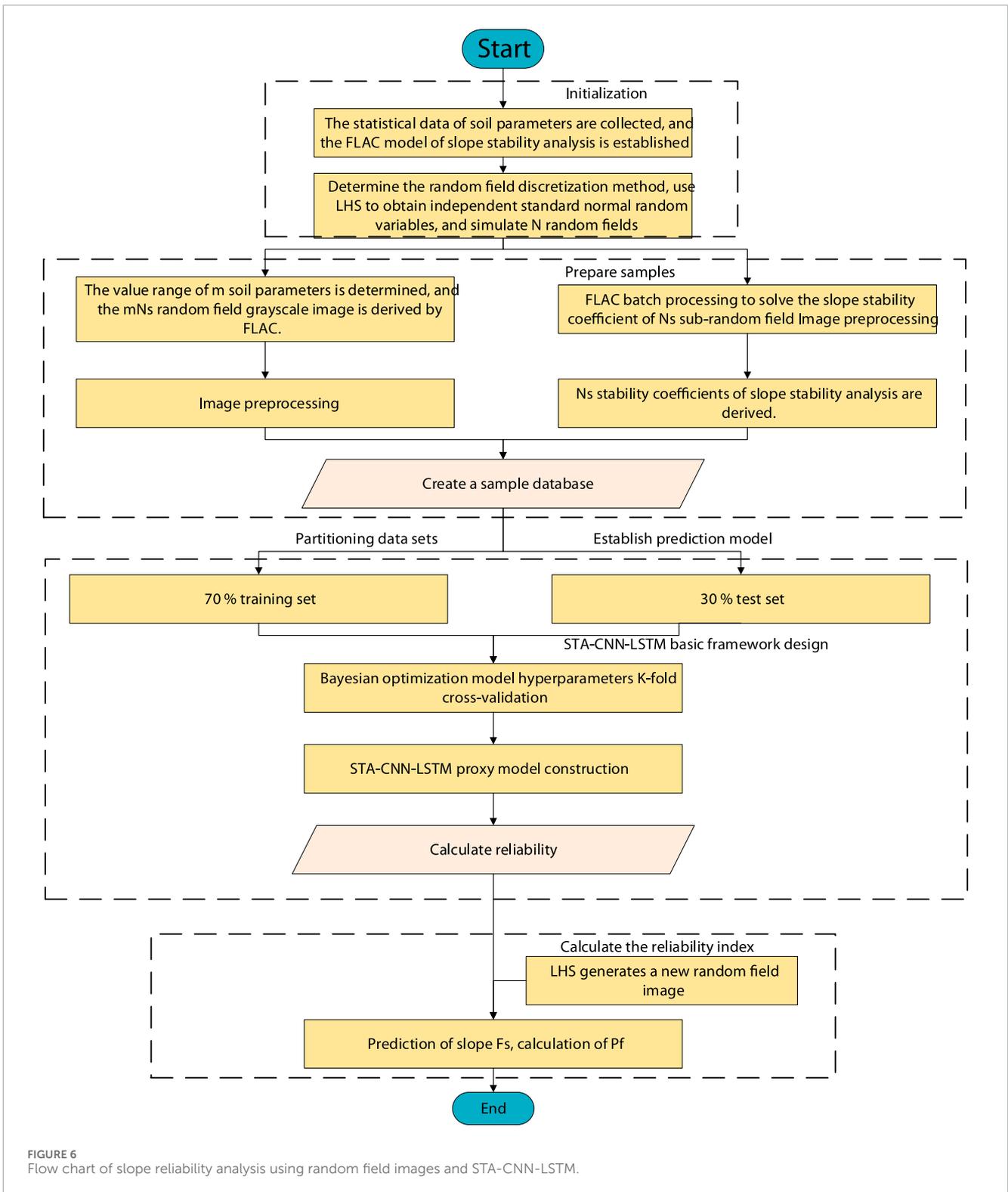
In Eq. 27, L_h and L_v respectively represent the horizontal and vertical lengths of the discretized domain of the random field.

3 Reliability analysis method for network models based on slope stochastic field digital images

3.1 Generation and preprocessing of stochastic field digital images

In the complex field of geotechnical engineering, transforming a discretized random field into a digital image represents an innovative approach. This process begins with the discretization of the random field, followed by its mapping onto digital images through a meticulously designed method. In this transformation, each coordinate of the random field's discretization point corresponds to a pixel in the image, with its value converted into the pixel's grayscale or color intensity, thereby creating a richly detailed visual representation. To preserve the spatial distribution characteristics of the random field, this study adopted a unique approach: initially mapping the random field onto the cell grid of a *FLAC^{3D}* model, then converting this data into a grayscale image. This method not only retains the spatial relationship of the random field but also sets the stage for subsequent machine learning analysis.

Once transformed, the random field images undergo a series of meticulous preprocessing steps before being inputted into the deep learning model. In the Python environment, we first convert the image from RGB mode to a more concise grayscale mode, emphasizing the image's structural features over color information. Next, to enhance processing efficiency, we remove excess blank areas from the image. Additionally, converting the data type is a crucial step, transforming the original uint8 data type (range 0–255) into a double-precision type (range 0–1) to meet the input requirements of deep learning models. Lastly, we adjust the image size to match the specifications of the CNN input layer. It should be noted that although rounding errors occur when scaling the random field to an integer range of 0–255, careful analysis shows that such errors are generally negligible and have minimal impact on model accuracy.



3.2 Reliability analysis based on simulation methods

After establishing the STA-CNN-LSTM surrogate model, simulation methods like the Monte Carlo Simulation (MCS) can be employed to calculate reliability indices. Latin Hypercube

Sampling (LHS) improves sampling efficiency, thus reducing computational load. Hence, this study uses LHS to generate samples. Slope stability analysis can be performed according to Eq. 28.

$$g(\mathbf{X}) = F_s(\mathbf{X}) - 1 \tag{28}$$

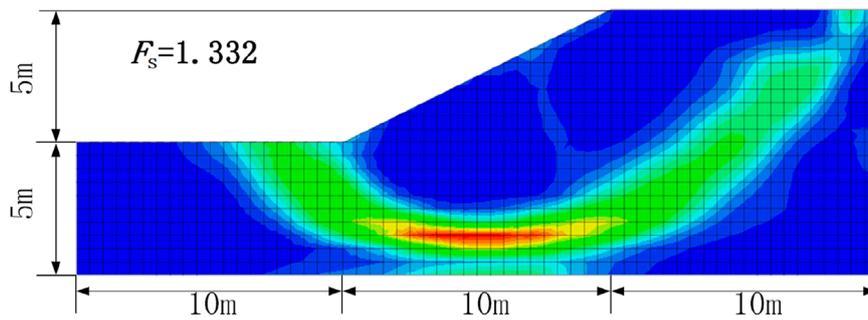


FIGURE 7
Deterministic analysis of a single layer clay slope under undrained condition.

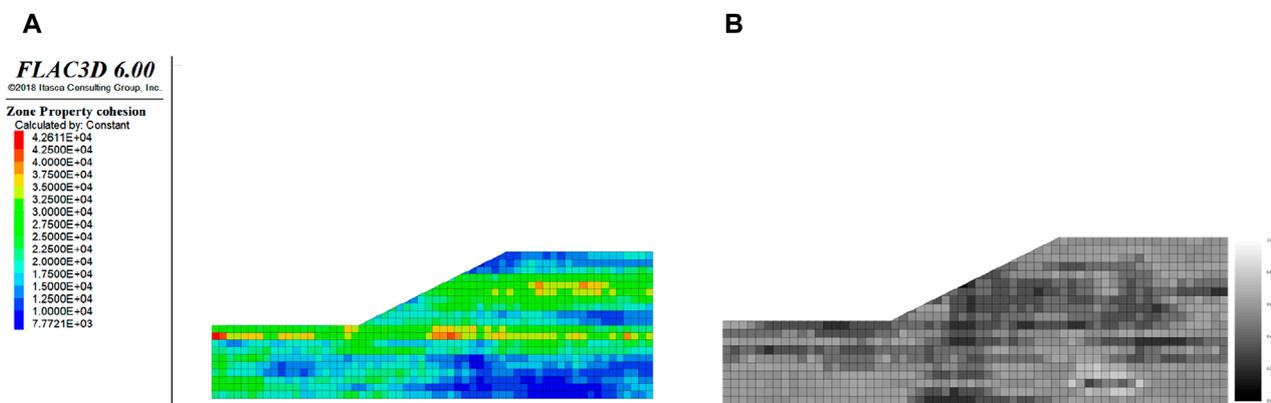


FIGURE 8
One realization of random fields (A) and corresponding gray image after preprocessing (B).

TABLE 1 Architecture of the CNN model.

Layer type	Number of filters	Filter size	Activation function	Output size
Input	-	-	-	12x40xNf
Convolutional	32	3x3	ReLU	12x40x32
Pooling	-	2x2	-	6x20x32
Convolutional	64	3x3	ReLU	6x20x64
Pooling	-	2x2	-	3x10x64
Fully Connected	-	-	ReLU	128
Regression Output	-	-	-	1 (MAE as loss)

In the formula, X represents the random variable of soil property parameters. The stochastic field images of the samples, after preprocessing, are directly processed by the CNN surrogate model to calculate the slope safety factor F_s , with the probability of failure P_f given by Eq. 29.

$$P_f = P(g(x) < 0) \approx \frac{n'(F_s(x) < 1)}{n} \quad (29)$$

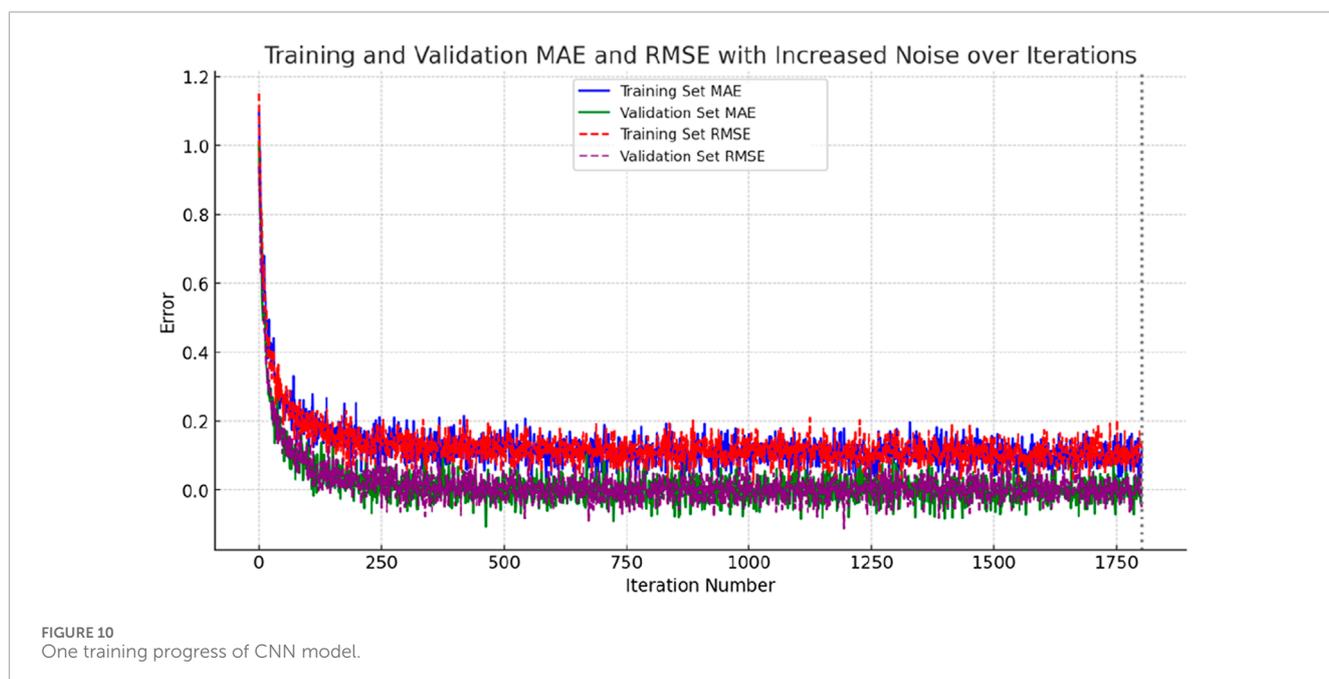
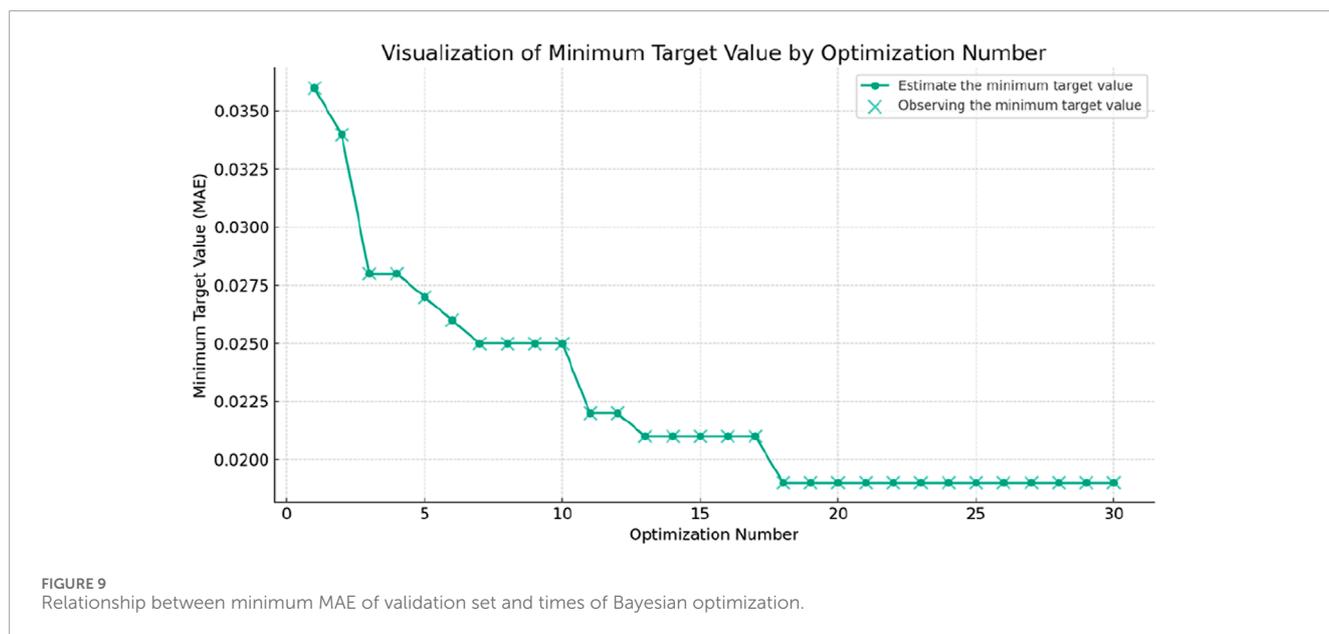
In this equation, n is the number of simulations, and $n'(F_s(x) < 1)$ represents the number of times F_s is less than 1. The corresponding reliability index β is approximately:

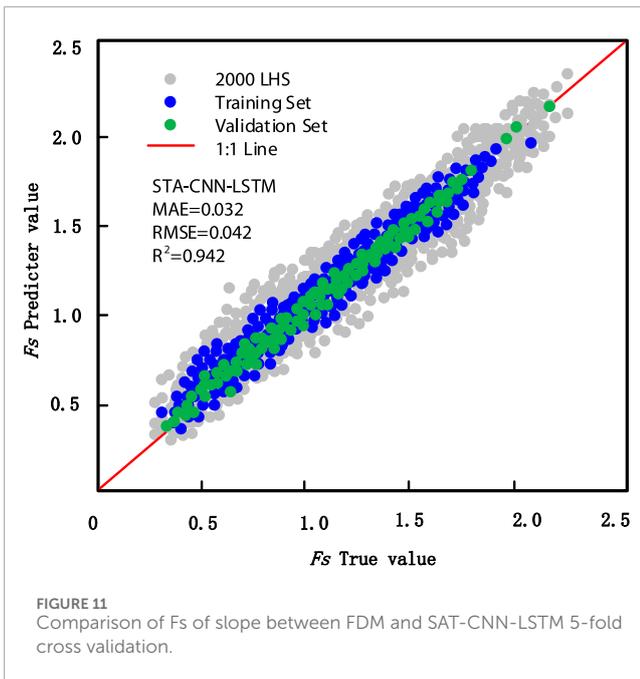
$$\beta \approx -\Phi^{-1}(P_f) \quad (30)$$

In Eq. 30, $\Phi^{-1}(P_f)$ represents the inverse function of the standard normal cumulative distribution function.

TABLE 2 Bayesian optimization of CNN hyperparameters.

Hyperparameters	Optimization range	Default value	Optimal hyperparameter
Initial Learning Rate	0.0001–0.1	0.001	0.0838
L2 Regularization Coefficient	0.00001–0.01	0.0001	0.0038
Dropout Rate	0–0.5	0.5	0.1157
Number of Filters Nf	2–256	64	16





3.3 Computational procedure and program implementation

The computational procedure of the method proposed in this paper is shown in Figure 6. This process includes four parts: establishing a numerical model considering the spatial variability of parameters (initialization), constructing a sample database, training the STA-CNN-LSTM surrogate model, and calculating reliability indices.

3.4 Case study verification

In the paper, an analysis is conducted on an undrained saturated clay slope, as shown in Figure 7. The slope has a height of 5 m and an angle of 26.6°. The undrained shear strength C_u is considered as a log-normal random field, with a mean value of 23 kPa and a coefficient of variation of 0.3. The horizontal and vertical autocorrelation distances are 20 m and 2 m, respectively, using an exponential autocorrelation function. The saturated unit weight of

the soil γ_{sat} is 20 kN/m³, the modulus of elasticity E is 100 MPa, the Poisson's ratio is 0.3, and the acceleration due to gravity is 9.8 m/s².

In our study, we utilized a $FLAC^{3D}$ model to precisely simulate the geological structure of an undrained saturated clay slope. The model was designed with a mesh cell edge length of 0.5 m, resulting in a total of 906 cells. This meshing strategy ensures that the mesh refinement level can capture geological details, adhering to a key criterion: the cell size should be less than half the correlation distance (i.e., the range of fluctuation) to ensure data accuracy (Chugh, 2002; Chugh et al., 2007). For the constitutive model, we adopted the ideal elastoplastic Mohr-Coulomb model combined with a non-associated flow rule, providing reliable predictions of mechanical behavior. With the undrained shear strength (C_u) parameter set to its mean value, our deterministic analysis yielded a slip coefficient (F_s) of 1.332, indicating slope stability.

To further analyze the stochastic field, we employed the Karhunen-Loève (KL) expansion method to discretize the C_u random field. Our findings revealed that increasing the number of expansion terms (M) from 100 to 200 significantly improved the discretization accuracy, with the ratio factor (ϵ) increasing from 0.92 to 0.95. This improvement highlights the importance of choosing an appropriate number of expansion terms for accurate representation of the stochastic field. In contrast to traditional surrogate models that rely on random variables, our approach directly inputs the stochastic field digital images into the STA-CNN-LSTM model, effectively overcoming the limitations of truncation terms in KL expansion. Consequently, we selected $M = 200$ to achieve a high discretization accuracy of $\epsilon = 0.95$ for the C_u random field.

Using independent standard normal random variables generated by the Latin Hypercube Sampling (LHS) method, we simulated 100 C_u random fields ($N_s = 100$). Figure 8 presents one realization of the C_u random field and its corresponding preprocessed grayscale image. In Figure 8A, the spatial distribution of the C_u random field is depicted. Based on this distribution, the calculated factor of safety (F_s) value is 0.963. Figure 8B displays the preprocessed grayscale image, sized $20 \times 60 \times 1$, with each grid representing a pixel. The pixel grayscale values range from 0 to 1, corresponding to C_u values in the interval [0, 60 kPa], covering 99.97% of possible values. These 100 samples will be utilized for training the STA-CNN-LSTM surrogate model, demonstrating the model's capability to predict slope stability accurately.

For the CNN model, a 10-layer architecture was employed. In the input layer, zero-center normalization was used, which involves subtracting the average value from the input data to reduce bias. The

TABLE 3 Comparison of reliability results of single-layer undrained clay slope.

Method	Model	Sample size	Fs mean	Fs standard deviation	Pf	β
LHS	FDM	2000	1.240	0.205	0.104	1.259
STA-CNN-LSTM-LHS	FDM	100	1.236	0.203	0.107	1.243
MCS	BSM	100,000	1.267	0.199	0.076	1.433
CNN-LSTM-LHS	FDM	100	1.237	0.204	0.110	1.240
CNN-LHS	FDM	100	1.239	0.205	0.114	1.237

filter size for the first convolutional layer was set to $9 \times 21 \times N_f$, where N_f is one of the key hyperparameters determined through Bayesian optimization. Since the ReLU activation function maintains the size of the feature map, the size of the feature map becomes $12 \times 40 \times N_f$ after the first convolution and activation. Subsequently, a 2×2 average pooling further reduces the feature map size, shrinking it to 25% of its original size, i.e., $6 \times 20 \times N_f$. In the second convolutional layer, the filter size is $3 \times 9 \times 2N_f$, doubling the number of filters from the first convolutional layer and further reducing the feature map size to $2 \times 6 \times 2N_f$. The subsequent part of the network includes dropout layers, fully connected layers, and a regression output layer with Mean Absolute Error (MAE) as the loss function, as shown in Table 1.

In terms of sample database usage, 30% of the data was randomly allocated as a validation set, with the remaining 70% used as the training set for Bayesian optimization of CNN hyperparameters. During the optimization process, the maximum number of training epochs for each hyperparameter combination was set to 3,000, and the maximum number of evaluations for different parameter combinations was limited to 30. This setting was aimed at reducing the time required for optimization. Table 2 details the optimization ranges for four hyperparameters, along with their default values and optimized best values. This approach not only improved the performance of the CNN model but also ensured efficient adjustment of the model to accommodate complex datasets.

Figure 9 shows the relationship between the minimum MAE on the validation set and the number of optimization iterations. Since the MAE of the validation set is fixed at the end of each optimization, the estimated minimum target value is equal to the observed value. The optimal hyperparameters listed in Table 1 are the results of the 18th optimization, corresponding to an MAE of 0.019, RMSE of 0.0238, and R^2 of 0.985 on the validation set, indicating high accuracy.

Training the STA-CNN-LSTM model with the optimal hyperparameters listed in Table 1, we employed a five-fold cross-validation approach to enhance its generalization capability. Figure 10 illustrates the training process of the STA-CNN-LSTM, showcasing the decreasing curves of the loss function values and RMSE for both the training and validation sets as the number of iterations increases. It is evident that the error for both datasets decreased rapidly and stabilized after the first 200 iterations. Due to the random dropout effect, the error in the training set is slightly higher than that in the validation set, as depicted in the figure. The error in the validation set gradually decreased and stabilized throughout the training process, with no overall upward trend, indicating that no overfitting occurred during training. Training was prematurely stopped after 1802 epochs, lasting approximately 2 min, demonstrating the efficiency of the STA-CNN-LSTM model's training time in this case.

Consequently, the STA-CNN-LSTM surrogate model has been successfully established. Building on this, slope reliability analysis was conducted for the case study. Initially, 2000 random fields (2000 LHS) were simulated using the Latin Hypercube Sampling (LHS) method, and a total of 2000 stochastic field grayscale images were generated using the image processing method proposed in this paper. These images, once predicted by the STA-CNN-LSTM surrogate model, yielded corresponding responses, i.e., the slope's

safety factor (Fs). Probability of failure (Pf) calculations for the slope were then performed based on this data.

To validate the method proposed in this paper, direct Finite Difference Method (FDM) stability analysis was performed on these 2000 simulated random field slopes using *FLAC*^{3D}, obtaining the 'true values' of Fs. Figure 10 compares the true values of Fs calculated by FDM and the predicted values of Fs calculated by the CNN model. The MAE, RMSE, and R^2 shown in the figure are the results calculated for the 2000 LHS data. In Figure 11, a few scatter points show predicted values greater than the true values, with some significant deviations. The maximum absolute error was 0.257 (for the point where the true value of Fs is just under 1.5), but overall, the model demonstrates high prediction accuracy.

Table 3 presents the results of the reliability analysis for a single-layer undrained saturated clay slope calculated by different methods. The Pf calculated directly by FDM for 2000 LHS simulations is 10.4%, with a 95% confidence interval of 9.06%–11.74%. The Pf calculated by the STA-CNN-LSTM-LHS method is 10.7%, which, compared to the former, results in a relative error of 2.88%. Simultaneously, the number of calculations for the STA-CNN-LSTM surrogate model is only 1/20th of the former. Since FDM calculations are time-consuming, the method proposed in this paper significantly reduces computation time. The Pf obtained using BSM-MCS is 7.6%, which differs considerably from the results of this paper, mainly because the Fs calculated by FDM is smaller than that by BSM, resulting in a relatively higher Pf. The failure rate for the STA-CNN-LSTM-LHS method is 10.7%, slightly higher for the CNN-LSTM-LHS method at 11.0%, and highest for the CNN-LHS method at 11.4%. This indicates that among these three methods, the STA-CNN-LSTM-LHS method performs the best in predicting the failure probability of slopes, while the CNN-LHS method performs the worst.

4 Conclusion

This study proposes a novel STA-CNN-LSTM model that incorporates a spatial-temporal attention mechanism for slope stability analysis in geotechnical engineering, particularly for the reliability analysis of single-layer undrained saturated clay slopes. By digitizing the stochastic field and applying deep learning technology, we transformed the slope's stochastic field characteristics into digital images and used a deep learning model to predict the slope's safety factor (Fs) and failure probability (Pf). Moreover, by optimizing the CNN model's hyperparameters through five-fold cross-validation and Bayesian optimization methods, the model's generalization ability and prediction accuracy were effectively enhanced, leading to the following conclusions:

- (1) For the case study in this paper, the CNN surrogate model accurately predicted the slope's safety factor, enabling a rapid and reliable assessment of slope stability. With the optimal hyperparameters based on Bayesian optimization, the STA-CNN-LSTM model achieved a Mean Absolute Error (MAE) of 0.019, a Root Mean Square Error (RMSE) of 0.0238, and a Coefficient of Determination (R^2) of 0.985 on the validation set.
- (2) The STA-CNN-LSTM surrogate model based on stochastic field digital images does not require dimensionality reduction

for high-dimensional problems, demonstrating independence from the method of stochastic field discretization and good versatility. The model's prediction results on 2000 simulated stochastic field samples showed high accuracy compared to the direct Finite Difference Method (FDM) stability analysis results, with a maximum absolute error of 0.257.

- (3) The *FLAC*^{3D} model accurately simulated the stochastic field and was compared with the direct Finite Difference Method (FDM) stability analysis results. The results showed that the STA-CNN-LSTM model had high accuracy in predicting the slope's safety factor, closely aligning with the 'true values' obtained from FDM calculations. Additionally, compared to traditional FDM methods, the STA-CNN-LSTM model significantly reduced computation time, proving its efficiency and feasibility in practical engineering applications.

Despite the excellent performance of the proposed STA-CNN-LSTM model in slope stability analysis, there are still some limitations and challenges. Future research can improve the model's generalization ability and robustness by introducing more diversified data sources and employing data augmentation techniques and explore more efficient network structures and algorithms to enhance the model's computational efficiency and prediction accuracy. Additionally, although the failure probability presented in the current case study is relatively high, the proposed method still has potential advantages in dealing with low failure probability situations. Future research can further explore and validate this point.

Data availability statement

The raw data supporting the conclusion of this article will be made available by the authors, without undue reservation.

Author contributions

NM: Data curation, Funding acquisition, Resources, Validation, Writing—original draft, Writing—review and

editing. ZY: Writing—review and editing, Visualization, Validation, Software, Methodology, Investigation, Data curation.

Funding

The author(s) declare that financial support was received for the research, authorship, and/or publication of this article. This work was supported by National Natural Science Foundation of Sichuan Province (2022NSFSC1128), National Natural Science Foundation of China (42207192), Scientific and Technological Research and Development Program of China National Railway Group Co., Ltd., (N2021G008), and Scientific and Technological Achievements Transformation Project of Qinghai Province (2022-SF-158). The authors declare that this study received funding from China National Railway Group Co., Ltd. The funder was not involved in the study design, collection, analysis, interpretation of data, the writing of this article, or the decision to submit it for publication.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

- Aminpour, M., Alaie, R., Khosravi, S., Kardani, N., Moridpour, S., and Nazem, M. (2023). Slope stability machine learning predictions on spatially variable random fields with and without factor of safety calculations. *Comput. Geotechnics* 153, 105094. doi:10.1016/j.compgeo.2022.105094
- Chakraborty, A. G. D., and Goswami, D. (2017). Prediction of slope stability using multiple linear regression (MLR) and artificial neural network (ANN). *Arabian J. Geosciences* 10 (17), 385. doi:10.1007/s12517-017-3167-x
- Chaudhary, M., Gastli, M. S., Nassar, L., and Karray, F. (2021). Deep learning approaches for forecasting strawberry yields and prices using satellite images and station-based soil parameters. doi:10.48550/arXiv.2102.09024
- Chugh, A. K. (2002). A method for locating critical slip surfaces in slope stability analysis: discussion. *Can. Geotech. J.* 39, 765–770. doi:10.1139/t02-042
- Chugh, A. K., Stark, T. D., and DeJong, K. A. (2007). Reanalysis of a municipal landfill slope failure near Cincinnati, Ohio, USA. *Can. Geotech. J.* 44 (1), 33–53. doi:10.1139/T06-089
- Ding, Y., Zhu, Y., Feng, J., Zhang, P., and Cheng, Z. (2020). Interpretable spatio-temporal attention LSTM model for flood forecasting. *Neurocomputing* 403, 348–359. doi:10.1016/j.neucom.2020.04.110
- Fang, K., Tang, H., Li, C., Su, X., An, P., and Sun, S. (2023). Centrifuge modelling of landslides and landslide hazard mitigation: a review. *Geosci. Front.* 14 (1), 101493. doi:10.1016/j.gsf.2022.101493
- Griffiths, D. V., and Lane, P. A. (1999). Slope stability analysis by finite elements. *Géotechnique* 49, 387–403. doi:10.1680/geot.1999.49.3.387
- He, X., Xu, H., and Sheng, D. (2023). Ready-to-use deep-learning surrogate models for problems with spatially variable inputs and outputs. *Acta Geotech.* 18 (4), 1681–1698. doi:10.1007/s11440-022-01706-2
- Li, W. C., Dai, F. C., Li, H. J., and Liu, H. D. (2008). Three-dimensional analysis on reinforcement effect of excavated rocky slope based on strength reduction technique. *J. Hydraulic Eng.* 39 (7), 877–882.
- Liu, Y. C., and Chen, C. S. (2007). A new approach for application of rock mass classification on rock slope stability assessment. *Eng. Geol.* 89 (1–2), 129–143. doi:10.1016/j.enggeo.2006.09.017
- Lu, N., and Godt, J. (2008). Infinite slope stability under steady unsaturated seepage conditions. *Water Resour. Res.* 44 (11). doi:10.1029/2008WR006976

- Ma, H., and Zhang, G. (2012). *Study on slope stability prediction based on LS-SVM*. Journal of Ningxia University (Natural Science Edition), 033(003), 250–253.
- Malkawi, A. I. H., Hassan, W. F., and Abdulla, F. A. (2000). Uncertainty and reliability analysis applied to slope stability. *Struct. Saf.* 22 (2), 161–187. doi:10.1016/S0167-4730(00)00006-0
- Poulos, H. G. (1995). Design of reinforcing piles to increase slope stability. *Can. Geotechnical J.* 32 (5), 808–818. doi:10.1139/t95-078
- Rahimi, M., Wang, Z., Shafieezadeh, A., Wood, D., and Kubatko, E. J. (2019). An adaptive kriging-based approach with weakly stationary random fields for soil slope reliability analysis. *Eighth Int. Conf. Case Hist. Geotechnical Eng.*, 148–157. Reston.
- Rosenbaum, P. L. S., and Rosenbaum, M. S. (2003). Artificial neural networks and grey systems for the prediction of slope stability. *Nat. Hazards* 30, 383–398. doi:10.1023/B:NHAZ.0000007168.00673.27
- Sameen, M. I., Pradhan, B., and Lee, S. (2020). Application of convolutional neural networks featuring Bayesian optimization for landslide susceptibility assessment. *Catena* 186, 104249. doi:10.1016/j.catena.2019.104249
- Shao, L. S., Ma, H., and Wen, T. X. (2015). Study on slope stability prediction based on RF-ELM model. *J. Saf. Sci. Technol* 11(3), 93–98. doi:10.11731/j.issn.1673-193x.2015.03.015
- Sidle, R. C. (1992). A theoretical model of the effects of timber harvesting on slope stability. *Water Resour. Res.* 28 (7), 1897–1910. doi:10.1029/92WR00804
- Tan, X. H., Dong, X. L., Fei, S. Z., Gong, W. P., Xiu, L. T., Hou, X. L., et al. (2020). Reliability analysis method based on KL expansion and its application. *Chin. J. Geotechnical Eng.* 42 (5), 808–816. (in Chinese).
- Wang, C., Wang, H., Qin, W., Wei, S., Tian, H., and Fang, K. (2023). Behaviour of pile-anchor reinforced landslides under varying water level, rainfall, and thrust load: insight from physical modelling. *Eng. Geol.* 325, 107293. doi:10.1016/j.enggeo.2023.107293
- Wang, Z. Z. (2022). Deep learning for geotechnical reliability analysis with multiple uncertainties. *J. Geotechnical Geoenvironmental Eng.* 148 (4), 06022001. doi:10.1061/(asce)gt.1943-5606.0002771
- Wang, Z. Z., and Goh, S. H. (2021). Novel approach to efficient slope reliability analysis in spatially variable soils. *Eng. Geol.* 281, 105989. doi:10.1016/j.enggeo.2020.105989
- Wu, W., and Sidle, R. C. (1995). A distributed slope stability model for steep forested basins. *Water Resour. Res.* 31, 2097–2110. doi:10.1029/95WR01136
- Xue, X., Yang, X., and Chen, X. (2014). Application of a support vector machine for prediction of slope stability. *Sci. China Technol. Sci. Engl. Ed.* 57, 2379–2386. doi:10.1007/s11431-014-5699-6
- Xu, G., Zhang, S., and Shi, W. (2023). *Instantaneous prediction of irregular ocean surface wave based on deep learning*. Ocean Engineering.
- Xue, X. (2016). Prediction of slope stability based on hybrid PSO and LSSVM. *J. Comput. Civ. Eng.* 31, 04016041. doi:10.1061/(ASCE)CP.1943-5487.0000607
- Zhang, F., Zhu, C., He, M., Dong, M., and Zhang, G. (2021). Failure mechanism and long short-term memory neural network model for landslide risk prediction. *Remote Sens.* 14, 166. doi:10.3390/rs14010166
- Zhang, L. Y., Zheng, Y. R., Zhao, S. Y., and Shi, W. M. (2003). The feasibility study of strength-reduction method with FEM for calculating safety factors of soil slope stability. *J. Hydraulic Eng.* 34 (1), 0021–0027. doi:10.1007/s11769-003-0044-1
- Zhang, Y., Chen, G., Zheng, L., Li, Y., and Zhuang, X. (2013). Effects of geometries on three-dimensional slope stability. *Can. Geotechnical J.* 50 (3), 233–249. doi:10.1139/cgj-2012-0279
- Zheng, H., Liu, D. F., and Li, C. G. (2010). Slope stability analysis based on elasto-plastic finite element method. *Int. J. Numer. Methods Eng.* 64 (14), 1871–1888. doi:10.1002/nme.1406