Inverse Analysis Of Internal Crack Propagation Depth In Cantilever Anti-Sliding Piles Based On Artificial Intelligence And XFEM

Qingyang Ren[1,2], Yanding Wang[1,2*], Songqiang Xiao[1,2], Yanping Jia[1,2], Senlin Gao[1,2], and Yong Zeng[1,2]

[1] *State Key Laboratory of Mountain Bridge and Tunnel Engineering, Chongqing Jiaotong University, Chongqing 400074, China*
[2] *School of Civil Engineering, Chongqing Jiaotong University, Chongqing 400074, China*
[*] *Corresponding author. E-mail: ydwang@mails.cqjtu.edu.cn*

Received: October 09, 2025; Accepted: April 19, 2026

Cantilever anti-sliding piles are critical for slope stability, yet their subsurface health assessment remains challenged due to the invisibility of internal damage. This study proposes a non-destructive inversion framework that integrates computer vision with computational mechanics to quantify internal crack propagation depth based on surface morphological features. To address the limitations of the standard Artificial Bee Colony (ABC) algorithm, specifically its slow convergence and susceptibility to local optima—an Improved ABC (I-ABC) algorithm is developed. This enhancement incorporates a neighborhood-based optimal guidance mechanism and an adaptive selection probability strategy to strike an optimal balance between global exploration and local exploitation. The proposed framework first employs an optimized YOLOv8 model, with hyper-parameters tuned via the I-ABC algorithm, to extract surface crack parameters. These parameters are subsequently mapped to internal depths using a normalized objective function derived from an Extended Finite Element Method (XFEM) forward modeling database. Experimental validation, utilizing a hybrid dataset comprising physical pile tests and numerical simulations, demonstrates that the proposed method achieves a crack depth inversion accuracy of 96%. Furthermore, statistical comparisons reveal that the I-ABC algorithm significantly outperforms Particle Swarm Optimization (PSO), Differential Evolution (DE), and Grey Wolf Optimizer (GWO) in terms of robustness and convergence speed. Ultimately, this framework provides a reliable and automated tool for the structural health monitoring of geotechnical infrastructure.

Keywords: Cantilever anti-sliding pile; Artificial Bee Colony Algorithm; Adaptive parameters; Crack propagation depth; Extended Finite Element Method

http://dx.doi.org/10.6180/jase.202609_32.028

1. Introduction

With the strategic deepening of China's Western Development program, the expansion of mountainous road networks has accelerated significantly, making slope stability under complex terrain a critical challenge. As an efficient landslide mitigation measure, anti-sliding piles utilize the soil arching effect to transfer sliding thrust to the pile bodies, thereby enhancing slope resistance. Compared to alternative retaining structures, anti-sliding piles offer superior load-bearing capacity with reduced material volume, making them indispensable in geotechnical reinforcement [1]. However, environmental and service conditions inevitably cause varying degrees of damage. Cracks, as a common defect in anti-sliding piles, directly impact their safety and durability. Current crack detection for anti-sliding piles relies primarily on manual methods, suffering from low efficiency, strong subjectivity, difficulty in quantifying crack morphology, and safety hazards. Therefore, there is an ur-

gent need to develop an automated crack detection method to overcome the limitations of manual inspection.

In recent years, the YOLOv8 neural network has occupied a pivotal position in image recognition tasks [2]. However, its algorithm may fail to clearly identify detection features for minute cracks on structural surfaces when applied to crack image recognition, leading to issues such as false positives and false negatives. With the widespread application of intelligent optimization algorithms in image recognition, a diverse array of algorithms has emerged, including Genetic Algorithm (GA) [3], Particle Swarm Optimization (PSO) [4], Ant Colony Optimization (ACO) [5], Grey Wolf Optimization (GWO) [6], and Artificial Bee Colony Algorithm (ABC) [7], have brought breakthrough progress to structural damage identification. Concurrently, advancements in computer technology and numerical simulation techniques have opened new avenues for non-destructive testing in structural engineering.

By integrating advanced numerical analysis techniques with actual measurement data, effective inversion analysis models can be constructed to identify specific parameters such as the location, size, and shape of internal structural defects [8]. The Extended Finite Element Method (XFEM) is a numerical approach for solving complex engineering problems. It introduces additional degrees of freedom and shape functions to the traditional finite element method to handle discontinuities like cracks and faults. Its core concept integrates the description of cracks or other discontinuities with the finite element model, enabling efficient capture of fracture processes and stress field changes without requiring remeshing [9]. The ABC algorithm draws inspiration from honeybee foraging behavior, simulating the individual division of labor and information sharing mechanisms during nectar collection. Compared to other optimization methods like genetic algorithms and ant colony optimization, ABC demonstrates significant advantages in optimization performance and algorithmic accuracy [10]. Furthermore, the algorithm features a simple structure, easy implementation, strong robustness, and fewer control parameters. Since its inception, ABC has attracted considerable attention from researchers and found extensive applications in path planning, layout optimization, and other fields. However, it also suffers from drawbacks such as slow convergence speed and susceptibility to local optima, leading to low solution accuracy. For instance, Bajer et al. [11] noted in their research that the ABC algorithm emphasizes exploration during the search phase but exhibits relatively weaker performance in exploitation. To overcome these deficiencies, this paper proposes an Improved ABC (I-ABC) algorithm by refining the search mechanisms and selection probability strategies. This enhancement increases the search diversity and prioritizes optimal nectar sources. By coupling the I-ABC-optimized YOLOv8 model with XFEM, this study establishes a comprehensive inversion framework to achieve precise quantification of internal crack propagation depth.

2. Theory and formula

2.1. Principles of the ABC Algorithm

The ABC Algorithm proposed by Karaboga in 2005, is an optimization algorithm that simulates the foraging behavior of bees. Bees are categorized into three functional groups: employed bees (performing local exploitation of current solutions), onlooker bees (selecting high-quality solutions based on shared information), and scout bees (conducting global exploration to discover new food sources) [7]. Nectar sources correspond to different defects, and the quality of nectar sources represents the fitness function value of defect parameters. The algorithm's strengths lie in its powerful search capability, minimal control parameters, straightforward implementation, and robust performance. The specific operational process comprises the following four phases:

1. Initialize the Population

   Initialize N nectar source positions $F_{ij,}$, where $i = 1,2,3,\ldots,N$ and $j = 1,2,3,\ldots,M$, using Eq. (1). $F_{i,j}$ denotes the j-dimensional position of nectar source i.

   $$F_{i,j} = F_j^{\min} + \left(F_j^{\max} - F_j^{\min}\right) \times \text{rand}(0,1) \quad (1)$$

   Where: j denotes the dimension of the search space; $F_j^{\max}$ and $F_j^{\min}$ represent the upper and lower bounds of the solution in the j-th dimension.

2. Employed Bee Phase

   Employed bees perform local searches within nearby domains based on current nectar source information. During the search process, let x_i denote the objective function value of the i-th feasible solution. The fitness value X_i of the i-th feasible solution is calculated using the following formula:

   $$X_i = \begin{cases} \frac{1}{1+x_i}, & x_i \geq 0 \\ 1 + |x_i|, & x_i < 0 \end{cases} \quad (2)$$

   When a forager locates a nectar source $F_{q,j}$ with higher nectar yield, it compares the fitness value X_i of the current source $F_{i,j,j}$ with the fitness value X_q of $F_{q,j}$, If $X_i > X_q$, $F_{q,j}$ is selected as the optimal nectar source.

The forager corresponding to the i-th source then locates the new nectar source $F'_{i,j}$ using the following formula:

$$F'_{i,j} = F_{i,j} + \left(F_{i,j} - F_{d,j}\right) \times \text{rand}(-1,1) \tag{3}$$

Where: $d \neq i$, the random number $\text{rand}(-1,1)$ generates new nectar sources $F'_{i,j}$ near the nectar source F_{ij}.

3. Onlooker Bee Phase

   Upon completing their search, employed bees share nectar information in the hive.

   Onlooker bees then choose a nectar source to follow based on a probability P_i, calculated according to the relative fitness of the sources:

$$P_i = \frac{X_i}{\sum_{i=1}^{N} X_i} \tag{4}$$

4. Scout Bee Phase

   If a nectar source fails to update to a high-quality source after reaching the iteration threshold, the nectar source $F_{i,j,j}$ will be discarded. The corresponding employed bee will transition to a scout bee, continuing to search for new nectar sources via Eq. (1). Upon finding a superior nectar source, it will revert to an employed bee.

Based on these four phases, the workflow of the ABC algorithm for solving optimization problems is illustrated in Fig. 1.

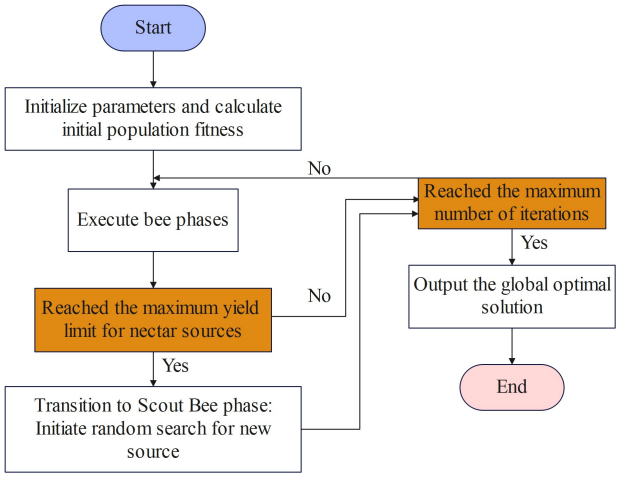


Fig. 1. ABC algorithm flowchart.

2.2. Improved ABC Algorithm

Traditional ABC algorithms typically employ random initialization to generate the initial population. While widely used, this approach often leads to slow convergence and a high susceptibility to local optima. To overcome these limitations, This study primarily focuses on enhancing the search mechanisms and selection probability strategies.

2.2.1. *Improvements to Search Mechanisms*

Zhu et al. [12] incorporated the global optimum solution $F_{\text{best},j}$ into the ABC algorithm, drawing on the update mechanism of Particle Swarm Optimization(PSO) to accelerate convergence. Gao et al. [13] further modified this by replacing the spatial upper bound with the global optimum. Although these modifications increase convergence speed, the risk of stagnating in local optima remains high. Therefore, this study introduces not only the global optimum solution but also the optimal solution within its neighborhood, denoted as $F_{\text{best},j,j}$. The improved search formula is as follows:

$$\begin{aligned} F_{i,j} = F_{best,i,j} &+ (F_{best,j} - F_{d,j}) \times \text{rand}(-1,1) \\ &+ (F_{best,j} - F_{i,j}) \times \text{rand}(0,1.5) \end{aligned} \tag{5}$$

From the above formula, it can be observed that incorporating the global optimal solution $F_{\text{best},i}$ and the neighborhood optimal solution $F_{\text{best},i,j}$ accelerates the algorithm's convergence rate while expanding its search capability and diversity.

2.2.2. *Improvement of Selection Probability*

In the standard ABC algorithm, the forager's selection primarily depends on the choice of optimal nectar sources and the application of the search equation. When selecting nectar sources using Eq. (4), significant differences in nectar source quality may lead to super-nectar sources with extremely high selection probabilities, causing the algorithm to converge to local optima. Conversely, minimal differences among nectar sources can degrade the algorithm into random selection, hindering convergence. Addressing the limitations of probabilistic selection methods in adapting to fitness value changes, Kong Depeng et al. [14] proposed a ranking-based inverse proportional selection strategy. Consider SN solutions $F_1, F_2, \ldots, F_{\text{SN}}$ with corresponding fitness values $q(F_1), q(F_2), \ldots, q(F_{\text{SN}})$. Sorting these values yields the sequence $F_{\text{R}(1)}, F_{\text{R}(2)}, \ldots, F_{\text{R}(\text{SN})}$, where $R(i)$ denotes the original sequence rank of the individual with fitness value rank $\underset{\sim}{i}$. The probability of an individual being selected as a follower is given by:

$$P'_i = \frac{1/i}{\sum_{i=1}^{SN} 1/i} \quad , \quad i = 1,2,\ldots,SN \tag{6}$$

As seen above, by constructing a selection mechanism through an inverse proportional function, individuals ranked higher in the sequence have a greater probability of being selected, while those ranked lower also retain a certain probability of selection. However, when nectar sources exhibit minimal differences, Eq. (5) struggles to reflect the true disparities between them—disparities that were previously captured before the improvement.

Therefore, this study proposes the following selection probability:

$$\bar{P}_i = \frac{P_i + P_i'}{2} \tag{7}$$

Where: P_i is the selection probability based on fitness; P_i' is the selection probability based on ranking.

2.3. Integrated Framework: YOLOv8 Optimization and Inversion Workflow

2.3.1. *I-ABC-based Hyper-parameter Optimization for YOLOv8*

In standard deep learning training, the selection of hyper-parameters (e.g., initial learning rate, weight decay, and momentum) significantly impacts model convergence and detection accuracy. Manual tuning is inefficient and prone to local optima. Therefore, this study employs the proposed Improved ABC (I-ABC) to adaptively search for the optimal hyper-parameter combination for YOLOv8.

Each nectar source $F_{i,j}$ represents a candidate vector:

$$V = [\eta, \lambda, \mu] \tag{8}$$

Where: η is the learning rate, λ is weight decay, and μ is momentum. The ranges are set as $\eta \in [10^{-5}, 10^{-1}]$, $\lambda \in [10^{-5}, 10^{-3}]$, and $\mu \in [0.8, 0.99]$.

The fitness value is defined as the Mean Average Precision (mAP @ 0.5) on the validation set after 100 training epochs:

$$X_i = mAP@0.5\,(\text{ Model }_i) \tag{9}$$

The Improved ABC described in Section 2.2 is used to update vector V. The YOLOv8 model is trained for a fixed few epochs (100 epochs) for each individual to evaluate fitness quickly.

2.3.2. *Overall Inversion Workflow and Objective Function Definition*

The proposed inversion system integrates the visual detection capabilities of YOLOv8 with the mechanical forward modeling of XFEM, optimized by the I-ABC algorithm. The workflow is redefined as a coupled iterative process, as illustrated in Fig. 2. This process comprises three distinct stages:

Phase 1. Surface Feature Extraction: The optimized YOLOv8 model detects cracks on the anti-sliding pile surface, outputting measured surface parameters vector W_{measured}, corresponding to the measured crack width, length, and area, respectively.

$$W_{\text{measured}} = [\text{ width,length,area }] \tag{10}$$

Phase 2. Forward Mapping Definition: A surrogate model database is constructed using XFEM. Let M_{XFEM} denote the forward operator mapping the internal crack depth d to the surface response vector W. For a given candidate depth d, the calculated surface feature vector is defined as:

$$W_{cal}(d) = M_{XFEM}(d) \tag{11}$$

Phase 3. Normalized Inverse Search: The I-ABC algorithm iteratively updates the candidate depth d to minimize the discrepancy between measured and calculated parameters.

To address the issue of dimensional inconsistency between crack width (mm), length (mm), and area (mm^2), a normalized least squares objective function is constructed. The fitness function $F(d)$ minimizes the relative error:

$$F(d) = \sum_{k=1}^{3} \omega_k \left(\frac{W_{\text{mea},k} - W_{\text{cal},k}(d)}{W_{\text{mea},k}} \right)^2 \tag{12}$$

Where: k is the dimension index (1 : width, 2 : length, 3 : area), ω_k is the weight coefficient for each parameter (set to 1.0 in this study to ensure equal contribution from all morphological features). The inversion process terminates when the fitness value falls below the tolerance threshold or the maximum iteration count is reached.

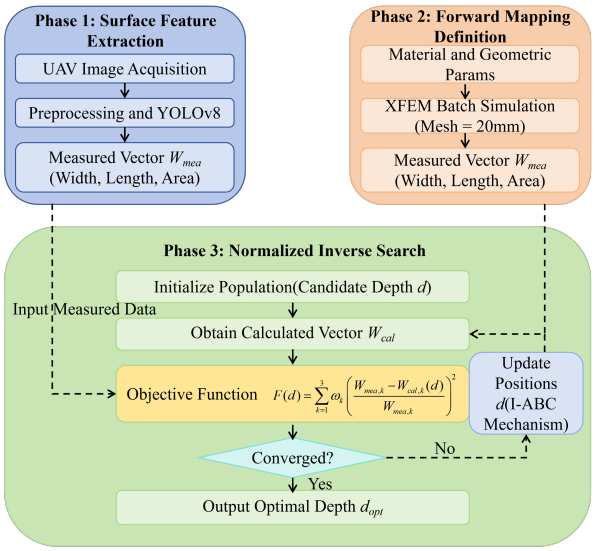


Fig. 2. Workflow Diagram.

3. Experimental setup

3.1. Construction of the Crack Dataset

3.1.1. Image Data Acquisition

A high-quality image dataset significantly enhances model training effectiveness and weight robustness. In the damage detection of cantilever anti-sliding piles, the performance of acquisition equipment and the design of image capture methods directly determine image quality [15]. This study employed a DJI drone capable of omnidirectional obstacle avoidance and Level 6 wind resistance to capture equidistant vertical images of surface cracks on concrete structures. Specific parameter information is detailed in Table 1.

3.1.2. Image Preprocessing

Image preprocessing is a critical step in computer vision. It involves adjusting and optimizing images captured in practical engineering through a series of processing techniques to enhance the accuracy and efficiency of subsequent tasks. We integrated the advantages of mean filtering, median filtering, Gaussian filtering, and complex algorithms based on wavelet transforms to perform image denoising. Histogram equalization combined with homomorphic filtering is employed to enhance target regions within the images. The comparison of preprocessing results is presented in Table 2. Since neural network training relies on large amounts of image data, this paper implements data augmentation based on OpenCV, employing random cropping, rotation, and random color transformations for data enhancement. To achieve crack detection, images require precise annotation to determine crack locations and morphology, providing data support for model training. LabelImg is selected as the annotation tool for this study. Designed specifically for object detection tasks, it supports multiple formats including Pascal VOC and YOLO, adapts to various machine learning frameworks, and features a user-friendly interface.

3.2. XFEM Mechanical Analysis and Modeling

Belytschko et al. [16] from Northwestern University pioneered the use of the XFEM for addressing cracks and other complex interface problems. In defect identification inversion models, XFEM decouples the shape function from the element topology through displacement discontinuity functions. The displacement pattern in XFEM can be expressed as:

$$u^h(x) = u^{FE} + U^{enr} \tag{13}$$

$$u^h(x) = \sum_{j=1}^{n} N_j(x)u_j + \sum_{k=1}^{m} N_k(x)\psi(x)a_k \tag{14}$$

$$\psi(x) = \text{sign}(\xi)\xi(x) \tag{15}$$

Where: N_j and N_k represent the shape functions associated with finite element (FE) and enriched (enr) nodes, respectively; u_j denotes the number of degrees of freedom in the conventional finite element method; a_k represents the additional enriched degrees of freedom; $\psi(x)$ is a set of enrichment functions associated with enriched nodes in the discontinuous domain; ξ is the perpendicular distance to the crack path; $\xi(x)$ is calculated as the derivative of the shape function.

Since ABAQUS's scripting interface is a data interface developed based on the Python language, integrating Python and ABAQUS forms a multi-source software integration platform to achieve the prediction of damage parameters for anti-sliding pile structures. This study simulates cracks in anti-sliding piles using XFEM. Selected parameters are listed in Table 3: the anchorage segment measures 400 mm , the loaded segment spans 700 mm , and the tension reinforcement consists of two Φ12 HRB400 bars (G1). The stirrups employ two Φ10 HRB400 bars (G2). One Φ8 HRB400 steel bar (G3) is used for the stirrups, with a spacing of 106 mm between stirrups. The concrete cover thickness is 20 mm . After a series of verification calculations, the reinforcement ratio of this cross-section meets the code requirements and will not result in under-reinforcement or over-reinforcement failure. The schematic diagram of the cantilever anti-sliding pile model structure is shown in Fig. 3.

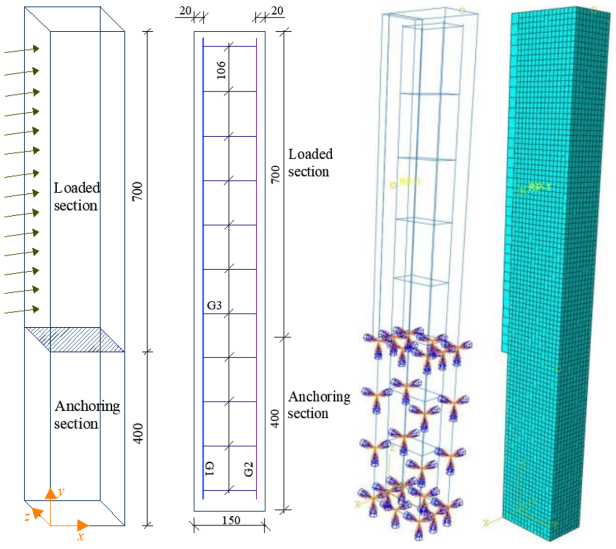


Fig. 3. Schematic diagram of cantilever anti-sliding pile model structure.

3.2.1. Validation of XFEM Forward Model and Mesh Convergence

Prior to generating the extensive training database, the reliability of the XFEM model was rigorously verified. A

Table 1. DJI drone specifications.

Brand/Model	DJI Air 3	Camera resolution	Wide-angle/Telephoto: 48 million
Duration	53 minutes	Maximum flight altitude	500 m
Image transmission distance	20 km	Sensor	Wide-angle/Telephoto: 1/1.3-inch
Anti wind grade	Level 6	Focal length/Aperture	Wide-angle : 24 mm F1.7 Telephoto : 70 mm F2.8

Table 2. Comparison of time performance and performance with different superpixel methods

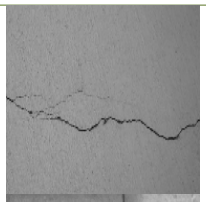
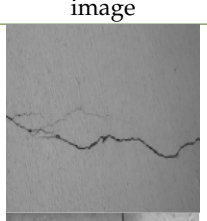
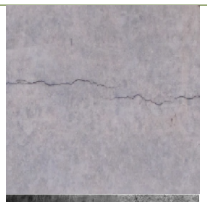
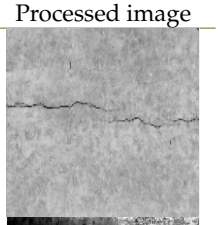
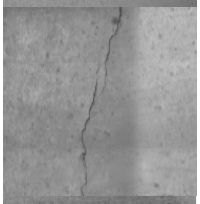
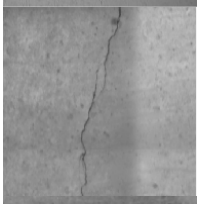
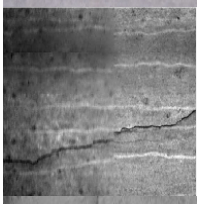
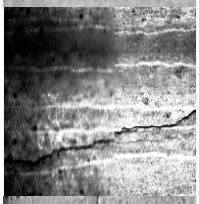
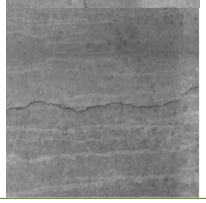
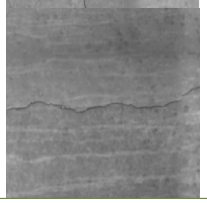
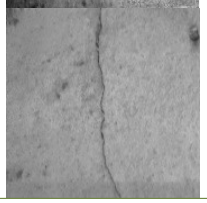
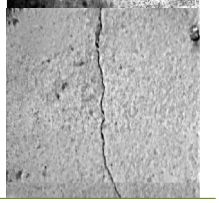
Preprocessing method	Original image	Processed image	Preprocessing method	Original image	Processed image
Mean-value			Wavelet denoising		
filtering			Histogram equalization		
Gaussian			Homomorphic filtering		

Table 3. Selected parameters for finite element modeling.

Name	Parameters
Anti-sliding pile dimensions	100 mm × 150 mm × 1100 mm
Concrete strength	C30
Concrete density	2500 kg/m^3
Concrete modulus of elasticity	6.5×10^3MPa
Concrete expansion angle ψ	35°
Concrete eccentricity	0.1
Concrete double-axis f_{b0} /Uniaxial compressive strength f_{c0}	1.16
Concrete unchanged stress ratio K	0.6667
Concrete cohesion coefficient	0.001
Length of tensioned reinforcing bars	1060 mm
Reinforcement density	7850 kg/m^3
Elastic modulus of steel reinforcement	200000 MPa
Steel poisson's ratio	0.3
Yield strength/Ultimate strength	400MPa/540MPa

mesh convergence study was conducted by varying the global mesh size from 10 mm to 50 mm. Results indicated that the variation in crack propagation paths and stress intensity factors stabilized when the mesh size was reduced below 25 mm. Consequently, a mesh size of 20 mm was selected for the database generation to achieve an optimal balance between computational accuracy and efficiency. The validation results of the numerical model are shown in Fig. 4.

Furthermore, the numerical model was validated against the experimental load-displacement curve of the undamaged Pile P1 (from the physical tests detailed in Section 4.2). The numerical simulation results demonstrated a strong correlation (R2 > 0.95) with the experimental data

during the linear elastic stage. This consistency confirms that the boundary conditions and material constitutive models adopted within the XFEM framework accurately reflect the physical behavior of cantilevered anti-sliding piles, generating a dataset of 24,000 samples required for subsequent inversion.

3.3. Crack Parameter Identification Test

3.3.1. Crack Extraction and Parameter Value Calculation

An improved algorithm optimizes the YOLOv8 network model for crack identification and localization, followed by extraction of crack morphological features and parameter information. A U2-Net convolutional neural network is employed for image segmentation (as shown in Fig. 5). The crack segmentation model utilizes 1×1 convolutional layers at both input and output ends to adjust the dimensions of the image and mask. The output is a mask image of the crack region, where pixels within the crack area are valued as 1 and all other regions as 0 (i.e., a foreground-as-1, background-as-0 image).

Crack parameters include width, length, and area. Crack width is defined as the pixel distance between any straight line passing through the crack and the intersection points of that line with the crack within the selected area [17]. The width calculation method based on the centerline obtains the crack edge line and centerline, then calculates the crack width using the spacing between intersection points where the centerline is perpendicular to the edge line. The crack parameter calculation is illustrated in Fig. 6.

Based on edge detection and skeleton extraction algorithms, pixel-level geometric feature parameters of the crack centerline are computed within the Region of Interest (RoI). Its length is calculated by accumulating the distances between adjacent points [18]. The distance L_D between adjacent pixels is expressed as follows:

$$L_D = \eta \times \sqrt{(x_{k+1} - x_k)^2 + (y_{k+1} - y_k)^2} \tag{16}$$

Where: η represents the pixel resolution; (x_k, y_k) and (x_{k+1}, y_{k+1}) denote the coordinate values of two adjacent points, respectively.

The crack area is calculated as the number of foreground pixels obtained through image post-processing, which represents the pixel area of the crack. The calculation method is as follows:

$$S = \int_S f^2(x,y)d_S \cong \sum f^2(x,y)d_S \tag{17}$$

Where: The curve S represents the crack region; d_S denotes the infinitesimal element of the crack area.

The crack parameter values calculated above are expressed in pixels. To convert these pixel values into actual parameters, a calibration algorithm can be employed. During the dataset construction phase, object distance parameters were recorded simultaneously. Using calibration blocks, the proportionality coefficient β between pixel size and physical size under different object distance conditions was calculated. Based on the β value, geometric parameters can be converted into actual values using the following calculation method:

$$W = \beta \times w \tag{18}$$

Where: W represents the actual geometric parameters of the crack, including width, length, or area; w denotes the pixel value of the crack image.

3.3.2. Accuracy Analysis of Crack Identification

To validate the accuracy of crack parameter identification, a test set comprising selected crack samples was used to compare detection values against actual measurement data. The specific comparison results are shown in Table 4. This demonstrates that the method can meet the requirements for detecting surface parameters of cracks in cantilever anti-sliding piles under practical conditions. Furthermore, it enables subsequent research into crack propagation depth.

4. Result discussions

4.1. Inversion Methodology

Inversion analysis is a robust technique for defect quantification [19], enabling the characterization of the size, geometry, and orientation of internal structural anomalies through non-destructive surface measurements. This is achieved by utilizing parameter information from key points on the structure's surface in conjunction with an effective inversion model. The solution framework for the inversion problem consists of two components: forward analysis and iterative minimization of the objective function [20]. The unknown parameters of the defect inversion model can be represented by the following vector set:

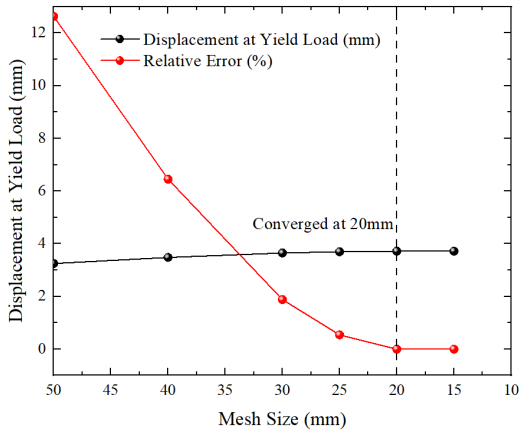
$$\beta = \{\beta_1, \beta_2, \beta_3, \ldots, \beta_n\} \in R^n \tag{19}$$

where: n denotes the number of parameters to be inverted.

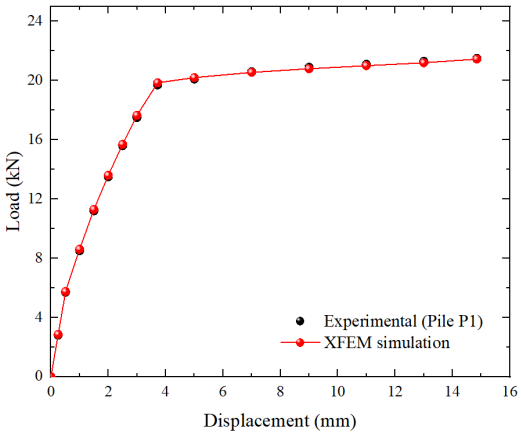
The objective function expression of the model is as follows:

$$\rho(\beta) = \frac{\sum_{i=1}^{N_T} \sum_{j=1}^{N_S} \left\| \alpha_{ij}^c(\beta) - \alpha_{ij}^m \right\|}{\sum_{i=1}^{N_T} \sum_{j=1}^{N_S} \left\| \alpha_{ij}^m \right\|} \tag{20}$$

Where: $\alpha_{ij}^c(\beta)$ and α_{ij}^m are the numerical calculation results and measured values at the detection points (displacement



(a) Mesh sensitivity analysis based on displacement at yield load.



(b) Comparison of load-displacement curves between the physical experiment (Pile P1) and the XFEM model.

Fig. 4. Validation of the numerical model.

Table 4. Comparison of crack detection results.

Crack image	Crack extraction	Crack length/mm		Crack area /mm^2		Average crack width /mm	
		Test value	Actual value	Test value	Actual value	Test value	Actual value
		351.2	357.6	885.5	892.3	2.493	2.469
		359.6	349.7	734.4	728.1	2.059	2.078
		409.3	406.5	869.4	875.2	2.147	2.153
		387.4	394.6	1092.7	1105.8	2.794	2.764

and strain in this paper); N_S denotes the number of detection points in the model pile; N_T denotes the number of measurement points in XFEM.

To facilitate computation, the inverse problem solution framework typically constructs an inverse form of the fitness function to enhance computational efficiency, expressed as follows:

$$f(\beta) = \frac{1}{\rho(\beta)} = \frac{\sum_{i=1}^{N_T}\sum_{j=1}^{N_S}\left\|\alpha_{ij}^{m}\right\|}{\sum_{i=1}^{N_T}\sum_{j=1}^{N_S}\left\|\alpha_{ij}^{c}(\beta)-\alpha_{ij}^{m}\right\|} \tag{21}$$

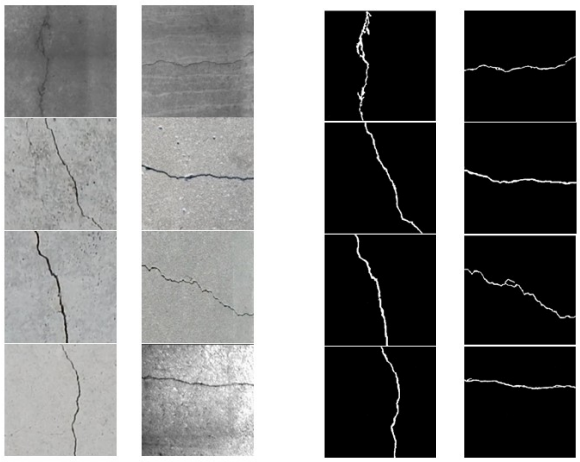
By examining Eqs. (20) and (21), it can be observed that as the value of the fitness function increases (i.e., the value of the objective function decreases), the quality of the solution is relatively higher.

4.2. Experimental Validation

To validate the practicality of the proposed I-ABC-YOLOv8 neural network for analyzing crack propagation in concrete structures, this study employed 24,000 samples generated through XFEM fundamental modeling combined with Python batch processing during the training phase. The sample set underwent independent testing using the I-ABC-YOLOv8 network model, the basic ABC-YOLOv8 network model, and the YOLOv8 network model.

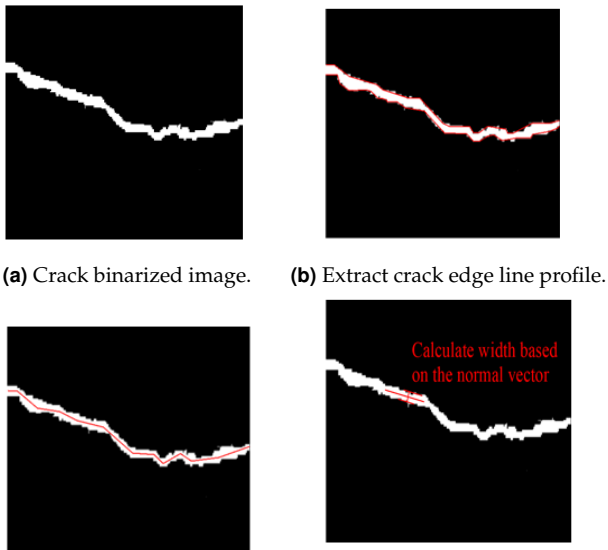
4.2.1. Verification of Hyper-parameter Optimization

To validate the efficacy of the I-ABC in optimizing YOLOv8 hyper-parameters, the training performance was monitored and compared against the standard configuration. As shown in Fig. 7. The optimization curve demonstrated



(a) Original image. **(b)** Segmentation results.

Fig. 5. Validation of the numerical model.



(a) Crack binarized image. **(b)** Extract crack edge line profile.

(c) Acquire crack image skeleton. **(d)** Calculate width based on the normal vector.

Fig. 6. Calculation of crack parameters for skid piles.

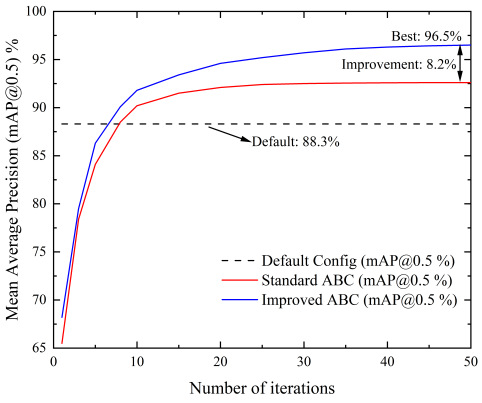


Fig. 7. Hyper-parameter optimization curve of YOLOv8.

that the I-ABC algorithm successfully identified a superior combination of initial learning rate, weight decay, and momentum within 50 iterations, effectively avoiding local optima that often trap manual tuning processes. The Mean Average Precision (mAP@0.5) of the YOLOv8 model optimized by I-ABC reached 96.5%, representing a substantial 8.2% improvement over the model using default hyperparameters. This optimization significantly enhanced the detection capability for minute surface cracks, providing high-quality input data for the subsequent inversion analysis.

4.2.2. *Experimental Design and Specimen Preparation*

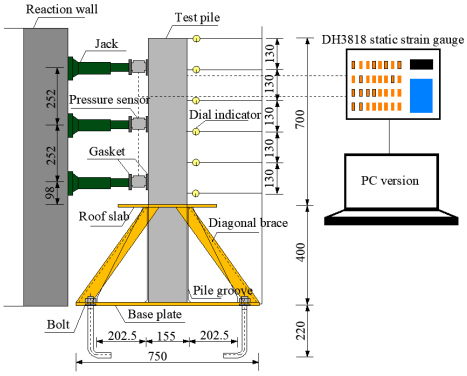
To ensure the experimental conditions align with engineering practice and the team's testing capabilities [21], the anchoring section of the anti-sliding pile was simulated using custom steel bearings. The horizontal thrust acting on the pile was treated as a uniformly distributed load according to the "Design Code for Landslide Prevention and Control" (GB/T 38509-2020) [22]. The experimental research was conducted in accordance with the "Standard Test Methods for Concrete Structures" (GB/T 50152-2012) [23]. The test apparatus and field test setup are shown in Fig. 8.

A total of ten test piles were fabricated for the experimental section, comprising five undamaged test piles and five test piles with initial cracks. The pile dimensions and internal reinforcement configurations were consistent with the finite element modeling parameters.

It should be noted that while the inversion model is trained on a comprehensive XFEM dataset comprising 24,000 scenarios, the 10 physical test piles serve specifically for experimental validation rather than training. In the field of large-scale civil engineering structures, destructive testing of proportional-scale anti-sliding piles is extremely costly and time-consuming. Therefore, a sample size of 10 specimens, covering various crack propagation stages, is considered statistically representative for validating the proposed method's engineering applicability and accuracy relative to the numerical model.

4.2.3. *Crack Propagation Testing of Specimens*

Fig. 9 shows the lateral displacement distribution of test pile P1 under different external loads. During load increment, the pile's horizontal deformation exhibited linear axial growth. The deformation progression can be cate-



(a) Schematic diagram of the test apparatus.

(b) Field view of test setup.

Fig. 8. Test pile operating system.

gorized into three distinct phases: crack-free operation, crack-bearing operation, and reinforcement yielding-pile failure. The failure characteristics are illustrated in Fig. 10.

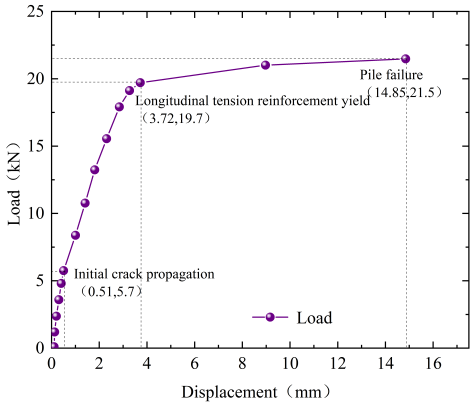


Fig. 9. Load-top lateral displacement distribution for test pile P1.

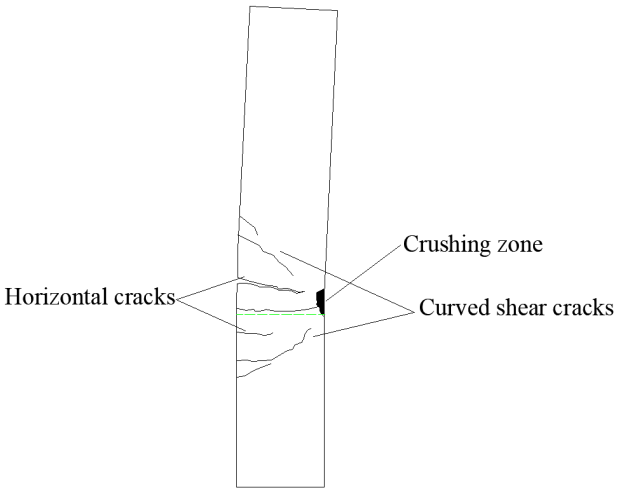


Fig. 10. P1 pile failure characteristics.

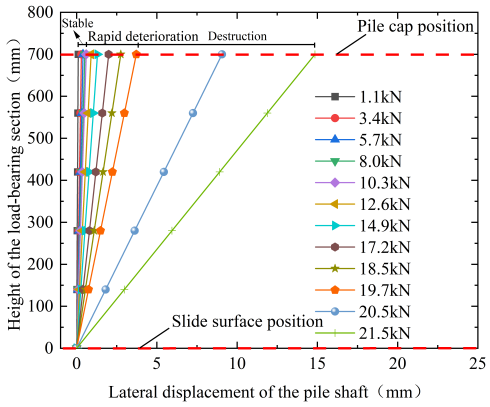


Fig. 11. Displacement distribution of test pile P1.

Fig. 11 shows the lateral displacement distribution of test pile P1 under different external loads. Test results indicate that the anti-sliding piles exhibit similar characteristics under various loading conditions, all displaying a linear gradient distribution pattern. Based on the evolution of top displacement, the structural working state can be divided into three phases: the stable period, the rapid deterioration period, and the failure period. Fig. 12 illustrates the evolution of concrete strain along the cross-sectional height of Test Pile P1 under varying loads. Under constant load conditions, the concrete strain values exhibit an approximate linear relationship with the radial distance from the neutral axis, and the strain gradient follows a linear distribution pattern. This conclusion generally satisfies the assumption of a plane section.

Table 5. Crack propagation depth inversion results.

Crack number	Crack I	Crack II	Crack III
Experimental measured values D_1(mm)	124.15	110.04	97.50
First inversion (mm)	125.02	109.38	96.47
Second inversion (mm)	124.94	109.45	96.44
Third inversion (mm)	124.87	109.51	96.43
Triple-reverse average D_A(mm)	124.94	109.45	96.45
Absolute error $D_E = \|D_A - D_1\|$ (mm)	0.97	0.59	1.05
Relative error $R_E = D_E/D_1$ (%)	0.78	0.54	1.08

Table 6. Performance comparison of different algorithms.

Algorithm	Avg. Iterations to Converge	Best Fitness (mm^2)	Worst Fitness (mm^2)	Std. Dev Std. Dev	Success Rate (<5% error)
PSO	20	1.240	2.312	0.54	75%
DE	52	0.785	2.18	0.12	83%
GWO	28	0.812	2.245	0.32	80%
Standard ABC	60	0.850	2.283	0.21	85%
Improved ABC	38	0.715	2.131	0.03	96%

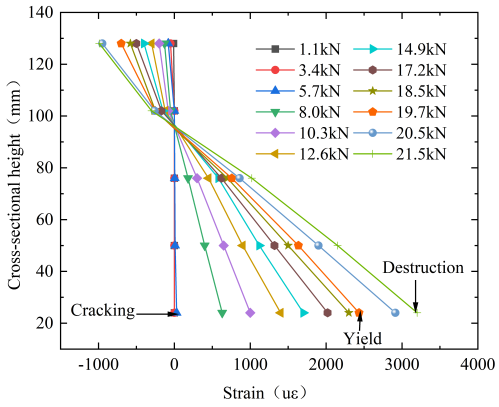


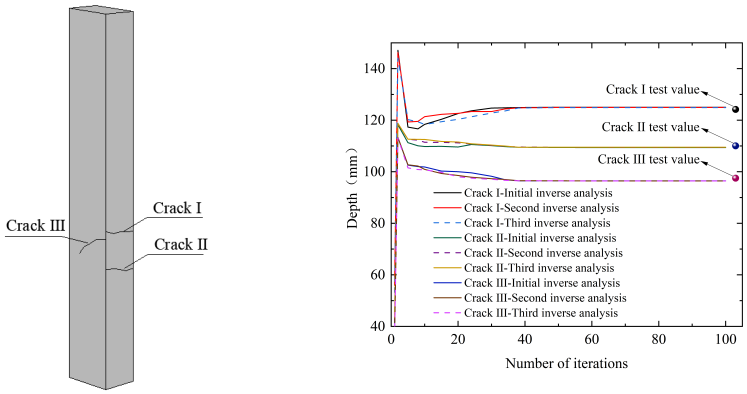
Fig. 12. Strain distribution at different heights in the concrete of test pile P1.

4.3. Inverse Optimization Analysis

Following the completion of the crack propagation tests on the test piles, three cracks were selected on the side of the test pile, designated as Crack I, Crack II, and Crack III. All three cracks were transverse cracks. Based on the proposed crack depth inversion analysis method, numerical simulations, and experimental data, the propagation depths of the three cracks were analyzed. Fig. 13(a) shows the locations of the three selected cracks on the test pile, while Fig. 13(b) presents the inversion results for their propagation depths. The iteration count on the horizontal axis reflects the convergence process of the inversion. The initial iteration results from the artificial bee colony algorithm exhibit randomness, but as the iteration count increases, the inversion results gradually approach the experimental values. To enhance the accuracy of the inversion results, the iteration count was set to 100, and the inversion process was repeated three times. The final result is the average of the three inversions.

As shown in Fig. 13, the propagation depths of all three cracks stabilized by the 38th iteration, with subsequent iterations yielding no further changes. Specific values are listed in Table 5. The absolute errors between the three inversion averages and the experimental values were 0.97 mm, 0.59 mm, and 1.05 mm, respectively, all controlled within 5%. The accuracy of the inversion analysis results meets the detection error requirements, validating the reliability of the inversion values for crack propagation depth in cantilever anti-sliding piles and the effectiveness of the inversion analysis method.

To further validate the reliability of the inversion results, Fig. 14 plots the variation curve of the fitness function value with the number of iterations during the crack propagation depth inversion process. As shown in Fig. 14, significant differences exist among the results during the first 15 iterations. However, with further iterations, all results ultimately converge to 0.715 mm^2. Theoretically, when the fitness function value converges to zero, the inversion results reach an optimal state. However, during the parameter inversion process, the microcracks generated inside the pile during the experimental loading phase were not considered. Furthermore, the cantilever anti-sliding pile component model constructed in the numerical analysis and the external boundary conditions of the numerically analyzed cantilever anti-sliding pile component model could not fully replicate the actual stress state of the component



(a) Crack selection. **(b)** Inversion results.

Fig. 13. Crack propagation depth inversion.

under laboratory or field conditions. This led to a certain deviation between the inversion results and the measured data. Nevertheless, the deviation between the values obtained from the inversion results and the actual values remained within a reasonable range, validating the effectiveness and reliability of the cantilever anti-sliding pile structure crack propagation depth inversion method proposed in this paper.

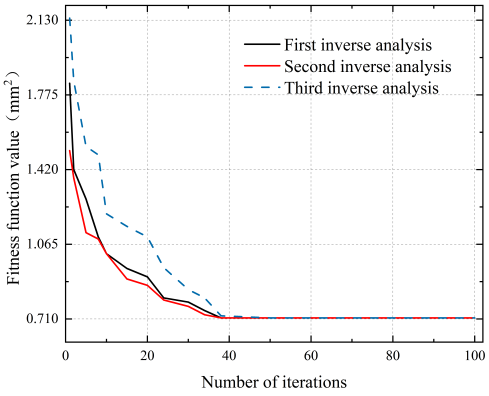


Fig. 14. Curve of fitness function values as a function of iteration count.

4.4. Statistical Performance Analysis and Algorithm Comparison

To demonstrate the superiority of the proposed I-ABC algorithm, specifically the contributions of the neighborhood optimal solution and adaptive selection probability, a comparative study was conducted against the Standard ABC Particle Swarm Optimization (PSO) Differential Evolution (DE) and the Grey Wolf Optimizer (GWO).

To ensure statistical significance and overcome the limitations of small-sample validation, a comprehensive validation set was constructed. This set comprises 52 actual morphological crack samples extracted from the 10 physical anti-sliding piles (spanning loading stages from initial cracking to failure) and a supplementary subset of 100 synthetic scenarios randomly sampled from the XFEM database (excluding the training set). This hybrid dataset design ensures that the algorithms are validated against both the complexity of real-world engineering noise and the theoretical diversity of numerical models.

Each of the five algorithms (I-ABC, ABC, PSO, DE, GWO) was run 30 independent times on the validation set. All algorithms were configured with a standardized population size of 50 and a maximum iteration count of 100 to ensure a fair comparison of computational efficiency.

Fig. 15 illustrates the representative convergence behavior of the five algorithms for a complex crack scenario. PSO exhibits rapid initial convergence but suffers from premature convergence, getting trapped in local optima around the 20th iteration with a fitness value of 1.24 mm². The DE algorithm prioritizes stability, achieving a low standard deviation of 0.12 at the expense of convergence speed (requiring 52 iterations). In contrast, the GWO improves exploration with a best fitness of 0.812 mm² but suffers from instability (standard deviation of 0.32), tending to oscillate around local optima. Standard ABC shows better exploration but slower convergence, stabilizing around the 60th iteration with a fitness value of 0.85 mm². Improved ABC achieves the best performance, converging to a minimal fitness value of 0.715 mm² by the 38th iteration. This confirms that introducing the global and neighborhood optimal guidance significantly enhances search efficiency.

As shown in Table 6, the Improved ABC not only offers

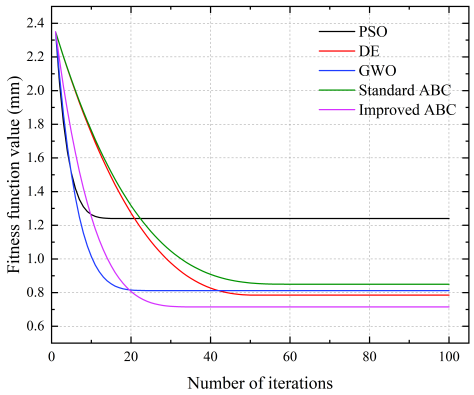


Fig. 15. Comparison of Three Algorithms.

higher accuracy but also greater stability (lowest standard deviation), The algorithm effectively resolves the complex nonlinear mapping problem in crack depth inversion.

5. Conclusions

This paper combines dynamic expansion finite elements with an improved ABC algorithm to propose an enhanced ABC-YOLOv8 network model capable of identifying surface cracks in anti-sliding piles. The I-ABC algorithm accelerates convergence speed, expands search capability and diversity, and increases the selection probability of optimal honeycomb sources.

1. The integration of the I-ABC algorithm with YOLOv8 effectively automated the hyper-parameter tuning process. The optimized model achieved a Mean Average Precision (mAP@0.5) of 96.5%, representing an 8.2% improvement over the default configuration, ensuring high-fidelity input data for subsequent mechanical analysis.

2. The proposed I-ABC algorithm overcomes the stagnation issues inherent in traditional meta-heuristics. Comparative statistical analysis across 30 independent runs demonstrates that I-ABC achieves a convergence success rate of 96% with a minimal standard deviation of 0.03. It statistically outperforms PSO, DE, and GWO, particularly in resolving complex non-linear mapping problems where standard algorithms often drift into local optima.

3. The multi-physics forward model, validated against experimental load-displacement curves (R2 > 0.95), provides a precise physical basis for inversion. The proposed normalized objective function successfully mitigates dimensional inconsistencies, yielding inverted crack depths with a maximum relative error of only 1.08% compared to physical measurements.

Acknowledgements

This work was financially supported by the Natural Science Foundation of Topics of National Key Research and Development Program of China (2023YFC3008300, 2023YFC3008304-4), National Natural Science Foundation of China (U20A20314).

References

[1] Z. Zhu and B. Lin, (2024) *"Calculations on Pile Spacing of Cantilever Anti-Slide Piles and Its Three-Dimensional Numerical Simulations"* **Chinese Journal of Industrial Construction *54***(10): 183–190. DOI: 10.13204/j.gyjzG22100503.

[2] Q. Ren, Y. Wang, and J. Shi, (2024) *"Advances in Target Detection Algorithms for Convolutional Neural Networks"* **Science Technology and Engineering *24***(32): 13665–13677. DOI: 10.12404/j.issn.1671-1815.2402976.

[3] J. H. Holland, (1992) *"Genetic Algorithms"* **Scientific American *267***(1): 66–73.

[4] J. Kennedy and R. Eberhart. "Particle swarm optimization". In: *Proceedings of ICNN'95 - International Conference on Neural Networks*. ***4***. 1995, 1942–1948 vol.4. DOI: 10.1109/ICNN.1995.488968.

[5] D. Marco, B. Mauro, and S. Thomas, (2006) *"Ant colony optimization"* **IEEE Computational Intelligence Magazine *1***(4): 28–39. DOI: 10.1109/MCI.2006.329691.

[6] S. Mirjalili, S. M. Mirjalili, and A. Lewis, (2014) *"Grey Wolf Optimizer"* **Advances in Engineering Software *69***: 46–61. DOI: 10.1016/j.advengsoft.2013.12.007.

[7] D. Karaboga and B. Basturk, (2007) *"A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm"* **Journal of Global Optimization *39***(3): 459–471. DOI: 10.1007/s10898-007-9149-x.

[8] X. Deng, C. Du, L. Jin, and X. Wang, (2019) *"Recognition of structural defects with improved artificial bee colony algorithm"* **Chinese Journal of Hefei University of Technology (Natural Science) *42***(06): 856–864. DOI: 10.3969/j.issn.1003-5060.2019.06.0123.

[9] Z. Xiang, L. Liu, and Z. Zhu, (2024) *"Research on Propagation Behaviors of Fatigue Cracks of Arc-Cutouts in Diaphragms Based on XFEM"* **Chinese Journal of Bridge Construction *54***(03): 46–53. DOI: 10.20051/j.issn.1003-4722.2024.03.007.

[10] B. Akay, D. Karaboga, B. Gorkemli, and E. Kaya, (2021) *"A survey on the Artificial Bee Colony algorithm variants for binary, integer and mixed integer programming problems"* **Applied Soft Computing *106***: 107351. DOI: 10.1016/j.asoc.2021.107351.

[11] D. Bajer and B. Zorić, (2019) *"An effective refined artificial bee colony algorithm for numerical optimisation"* **Information Sciences *504***: 221–275. DOI: 10.1016/j.ins.2019.07.022.

[12] G. Zhu and S. Kwong, (2010) *"Gbest-guided artificial bee colony algorithm for numerical function optimization"* **Applied Mathematics and Computation *217***(7): 3166–3173. DOI: 10.1016/j.amc.2010.08.049.

[13] W.-f. Gao and S.-y. Liu, (2012) *"A modified artificial bee colony algorithm"* **Computers & Operations Research *39***(3): 687–697. DOI: 10.1016/j.cor.2011.06.007.

[14] D. Kong, T. Chan, W. Dai, Q. Wang, and H. Sun, (2019) *"An improved artificial bee colony algorithm based on the ranking selection and the elite guidance"* **Chinese Journal of Control and Decision *34***(04): 781–786. DOI: 10.13195/j.kzyjc.2017.1334.

[15] D. Guo. *"Research and Application of Hidden Trouble Detection Model of Transmission Line Based on Improved YOLOX"*. (mathesis). Liaoning Technical University,CHINA, 2023. DOI: 10.27210/d.cnki.glnju.2023.000583.

[16] T. Belytschko and T. Black, (1999) *"Elastic crack growth in finite elements with minimal remeshing"* **International Journal for Numerical Methods in Engineering *45***(5): 601–620. DOI: 10.1002/(SICI)1097-0207(19990620)45:5<601::AID-NME598>3.0.CO;2-S.

[17] J. Yu, X. X. Feng Li, P. Zhu, X. Wu, and P. Lu, (2021) *"Intelligent Identification of Bridge Structural Cracks Based on Unmanned Aerial Vehicle and Mask R-CNN"* **Chinese Journal of Highway and Transport *34***(12): 80–90. DOI: 10.19721/j.cnki.1001-7372.2021.12.007.

[18] Z. Fang, J. Xia, and C. Liu, (2016) *"Crack shape detection on the structural surface based on image analysis technology"* **Chinese Journal of Railway Science and Engineering *13***(12): 2447–2454. DOI: 10.19713/j.cnki.43-1423/u.2016.12.019.

[19] S. Hao, W. Haim, and B. Raimondo, (2013) *"Nondestructive identification of multiple flaws using XFEM and a topologically adapting artificial bee colony algorithm"* **International Journal for Numerical Methods in Engineering *95***(10): 871–900. DOI: 10.1002/nme.4529.

[20] J. Song. *"Research of identification method for bridge local damage based on deep learning and optimal inversion analysis"*. (phdthesis). Tianjin University,CHINA, 2020. DOI: 10.27356/d.cnki.gtjdu.2020.002921.

[21] H. Jin. *"Study on the Service Performance and Durability Limit Life of Cantilever Anti-sliding Pile under Simulated Acid Rain Erosion Environment"*. (phdthesis). Chongqing Jiaotong University,CHINA, 2024. DOI: 10.27671/d.cnki.gcjtc.2024.000015.

[22] GB/T38509-2020. *Code for the Design of Landslide Stabilization*. 2020.

[23] G. 50152-2012. *Standard for test method of concrete structures*. 2012.