Artificial Intelligence Applications in Civil Engineering 2020

Lead Guest Editor: Tayfun Dede
Guest Editors: Murat Kankal, Ali Reza Vosoughi, Maksym Grzywinski, and Moacir Kripka

Artificial Intelligence Applications in Civil Engineering 2020

Artificial Intelligence Applications in Civil Engineering 2020

Lead Guest Editor: Tayfun Dede
Guest Editors: Murat Kankal, Ali Reza Vosoughi, Maksym Grzywinski, and Moacir Kripka

Zhaofei Chu, China
Wonseok Chung, Republic of Korea
Donato Ciampa, Italy
Gian Paolo Cimellaro, Italy
Francesco Colangelo, Italy
Romulus Costache, Romania
Liviu-Adrian Cotfas, Romania
Antonio Maria D'Altri, Italy
Bruno Dal Lago, Italy
Amos Darko, Hong Kong
Arka Jyoti Das, India
Dario De Domenico, Italy
Gianmarco De Felice, Italy
Stefano De Miranda, Italy
Maria T. De Risi, Italy
Tayfun Dede, Turkey
Sadik O. Degertekin, Turkey
Camelia Delcea, Romania
Cristoforo Demartino, China
Giuseppe Di Filippo, Italy
Luigi Di Sarno, Italy
Fabio Di Trapani, Italy
Aboelkasim Diab, Egypt
Thi My Dung Do, Vietnam
Giulio Dondi, Italy
Jiangfeng Dong, China
Chao Dou, China
Mario D'Aniello, Italy
Jingtao Du, China
Ahmed Elghazouli, United Kingdom
Francesco Fabbrocino, Italy
Flora Faleschini, Italy
Dingqiang Fan, Hong Kong
Xueping Fan, China
Qian Fang, China
Salar Farahmand-Tabar, Iran
Ilenia Farina, Italy
Roberto Fedele, Italy
Guang-Liang Feng, China
Luigi Fenu, Italy
Tiago Ferreira, Portugal
Marco Filippo Ferrotto, Italy
Antonio Formisano, Italy
Guoyang Fu, Australia
Stefano Galassi, Italy

Junfeng Gao, China
Meng Gao, China
Giovanni Garcea, Italy
Enrique García-Macías, Spain
Emilio García-Taengua, United Kingdom
DongDong Ge, USA
Khaled Ghaedi, Malaysia
Khaled Ghaedi, Malaysia
Gian Felice Giaccu, Italy
Agathoklis Giaralis, United Kingdom
Ravindran Gobinath, India
Rodrigo Gonçalves, Portugal
Peilin Gong, China
Belén González-Fonteboa, Spain
Salvatore Grasso, Italy
Fan Gu, USA
Erhan Güneyisi, Turkey
Esra Mete Güneyisi, Turkey
Pingye Guo, China
Ankit Gupta, India
Federico Gusella, Italy
Kemal Hacıefendioğlu, Turkey
Jianyong Han, China
Song Han, China
Asad Hanif, Macau
Hadi Hasanzadehshooiili, Canada
Mostafa Fahmi Hassanein, Egypt
Amir Ahmad Hedayat, Iran
Khandaker Hossain, Canada
Zahid Hossain, USA
Chao Hou, China
Biao Hu, China
Jiang Hu, China
Xiaodong Hu, China
Lei Huang, China
Cun Hui, China
Bon-Gang Hwang, Singapore
Jijo James, India
Abbas Fadhil Jasim, Iraq
Ahad Javanmardi, China
Krishnan Prabhakan Jaya, India
Dong-Sheng Jeng, Australia
Han-Yong Jeon, Republic of Korea
Pengjiao Jia, China
Shaohua Jiang, China

Chayut Ngamkhanong (iD), Thailand
Trung Ngo, Australia
Tengfei Nian, China
Mehdi Nikoo (iD), Canada
Youjun Ning (iD), China
Olugbenga Timo Oladinrin (iD), United Kingdom
Oladimeji Benedict Olalusi, South Africa
Timothy O. Olawumi (iD), Hong Kong
Alejandro Orfila (iD), Spain
Maurizio Orlando (iD), Italy
Siti Aminah Osman, Malaysia
Walid Oueslati (iD), Tunisia
SUVASH PAUL (iD), Bangladesh
John-Paris Pantouvakis (iD), Greece
Fabrizio Paolacci (iD), Italy
Giuseppina Pappalardo (iD), Italy
Fulvio Parisi (iD), Italy
Dimitrios G. Pavlou (iD), Norway
Daniele Pellegrini (iD), Italy
Gatheeshgar Perampalam (iD), United Kingdom
Daniele Perrone (iD), Italy
Giuseppe Piccardo (iD), Italy
Vagelis Plevris (iD), Qatar
Andrea Pranno (iD), Italy
Adolfo Preciado (iD), Mexico
Chongchong Qi (iD), China
Yu Qian, USA
Ying Qin (iD), China
Giuseppe Quaranta (iD), Italy
Krishanu ROY (iD), New Zealand
Vlastimir Radonjanin, Serbia
Carlo Rainieri (iD), Italy
Rahul V. Ralegaonkar, India
Raizal Saifulnaz Muhammad Rashid, Malaysia
Alessandro Rasulo (iD), Italy
Chonghong Ren (iD), China
Qing-Xin Ren, China
Dimitris Rizos (iD), USA
Geoffrey W. Rodgers (iD), New Zealand
Pier Paolo Rossi, Italy
Nicola Ruggieri (iD), Italy
JUNLONG SHANG, Singapore

Nikhil Saboo, India
Anna Saetta, Italy
Juan Sagaseta (iD), United Kingdom
Timo Saksala, Finland
Mostafa Salari, Canada
Ginevra Salerno (iD), Italy
Evangelos J. Sapountzakis (iD), Greece
Vassilis Sarhosis (iD), United Kingdom
Navaratnarajah Sathiparan (iD), Sri Lanka
Fabrizio Scozzese (iD), Italy
Halil Sezen (iD), USA
Payam Shafigh (iD), Malaysia
M. Shahria Alam, Canada
Yi Shan, China
Hussein Sharaf, Iraq
Mostafa Sharifzadeh, Australia
Sanjay Kumar Shukla, Australia
Amir Si Larbi (iD), France
Okan Sirin (iD), Qatar
Piotr Smarzewski (iD), Poland
Francesca Sollecito (iD), Italy
Rui Song (iD), China
Tian-Yi Song, Australia
Flavio Stochino (iD), Italy
Mayank Sukhija (iD), USA
Piti Sukontasukkul (iD), Thailand
Jianping Sun, Singapore
Xiao Sun (iD), China
T. Tafsirojjaman (iD), Australia
Fujiao Tang (iD), China
Patrick W.C. Tang (iD), Australia
Zhi Cheng Tang (iD), China
Weerachart Tangchirapat (iD), Thailand
Xiaxin Tao, China
Piergiorgio Tataranni (iD), Italy
Elisabete Teixeira (iD), Portugal
Jorge Iván Tobón (iD), Colombia
Jing-Zhong Tong, China
Francesco Trentadue (iD), Italy
Antonello Troncone, Italy
Majbah Uddin (iD), USA
Tariq Umar (iD), United Kingdom
Muahmmad Usman, United Kingdom
Muhammad Usman (iD), Pakistan
Mucteba Uysal (iD), Turkey

Ilaria Venanzi, Italy
Castorina S. Vieira, Portugal
Valeria Vignali, Italy
Claudia Vitone, Italy
Liwei WEN, China
Chunfeng Wan, China
Hua-Ping Wan, China
Roman Wan-Wendner, Austria
Chaohui Wang, China
Hao Wang, USA
Shiming Wang, China
Wayne Yu Wang, United Kingdom
Wen-Da Wang, China
Xing Wang, China
Xiuling Wang, China
Zhenjun Wang, China
Xin-Jiang Wei, China
Tao Wen, China
Weiping Wen, China
Lei Weng, China
Chao Wu, United Kingdom
Jiangyu Wu, China
Wangjie Wu, China
Wenbing Wu, China
Zhixing Xiao, China
Gang Xu, China
Jian Xu, China
Panpan, China
Rongchao Xu, China
HE YONGLIANG, China
Michael Yam, Hong Kong
Hailu Yang, China
Xu-Xu Yang, China
Hui Yao, China
Xinyu Ye, China
Zhoujing Ye, China
Gürol Yildirim, Turkey
Dawei Yin, China
Doo-Yeol Yoo, Republic of Korea
Zhanping You, USA
Afshar A. Yousefi, Iran
Xinbao Yu, USA
Dongdong Yuan, China
Geun Y. Yun, Republic of Korea

Hyun-Do Yun, Republic of Korea
Cemal YİĞİT, Turkey
Paolo Zampieri, Italy
Giulio Zani, Italy
Mariano Angelo Zanini, Italy
Zhixiong Zeng, Hong Kong
Mustafa Zeybek, Turkey
Henglong Zhang, China
Jiupeng Zhang, China
Tingting Zhang, China
Zengping Zhang, China
Zetian Zhang, China
Zhigang Zhang, China
Zhipeng Zhao, Japan
Jun Zhao, China
Annan Zhou, Australia
Jia-wen Zhou, China
Hai-Tao Zhu, China
Peng Zhu, China
QuanJie Zhu, China
Wenjun Zhu, China
Marco Zucca, Italy
Haoran Zuo, Australia
Junqing Zuo, China
Robert Černý, Czech Republic
Süleyman İpek, Turkey

Contents

Research Article

Research on Deformation Prediction of Foundation Pit Based on PSO-GM-BP Model

Dongge Cui ⬡,[1] **Chuanqu Zhu** ⬡,[1] **Qingfeng Li** ⬡,[2] **Qiyun Huang** ⬡,[1] and **Qi Luo** ⬡[1]

[1]*School of Resource & Environment and Safety Engineering, Hunan University of Science and Technology, Xiangtan, Hunan 411201, China*
[2]*Institute of Mineral Engineering, Hunan University of Science and Technology, Xiangtan, Hunan 411201, China*

Correspondence should be addressed to Dongge Cui; 170101040001@mail.hnust.edu.cn

Deformation prediction is significant to the safety of foundation pits. Against with low accuracy and limited applicability of a single model in forecasting, a PSO-GM-BP model was established, which used the PSO optimization algorithm to optimize and improve the GM (1, 1) model and the BP network model, respectively. Combining a small amount of measured data during the excavation of a bottomless foundation pit in a Changsha subway station, the calculations based on the PSO-GM model, the PSO-BP network model, and the PSO-GM-BP model compared. The results show that both the GM (1, 1) and BP neural network models can predict accurate results. The prediction optimized by the particle swarm algorithm is more accurate and has more substantial applicability. Due to its reliable accuracy and wide application range, the PSO-GM-BP model can effectively guide the construction of foundation pits, and it also has certain reference significance for other engineering applications.

1. Introduction

In-depth foundation pit engineering is a general term for a series of work carried out to ensure deep foundation pit construction safety and the surrounding environment not harmed. Safety construction and monitoring and early warning are also included [1]. Since the foundation pit's design cannot be entirely consistent with the actual situation, the construction conditions are complex and changeable. The environment of the foundation pit is also for various reasons. During the standard construction of the foundation pit, some uncontrollable conditions will also occur. When the deformation is severe, significant accidents such as the foundation pit's overall instability and the collapse of surrounding buildings may occur [2–4]. The purpose of foundation pit deformation monitoring is to ensure the smooth construction of foundation pit engineering so that the foundation pit deformation is within a safe and controllable range. Existing deformation evaluation indicators compare the amount of change and control value instead of using a more reasonable model to predict the

foundation pit's deformation status to grasp the foundation pit's further development trend. So far, we have many mature deformation analysis methods. The more common ones are the regression method, time series analysis model, gray system analysis model, Kalman filter model, artificial neural network model, spectrum analysis method, etc. [5–10]. The application of machine learning methods in different engineering fields is becoming more and more extensive [11].

The deformation process of the foundation pit is an uncertain system with many factors and complicated construction conditions. Therefore, it could be regarded as a gray information system [12]. The GM (1, 1) model can extract the chaotic data series. Trends, generate new data columns and use them for predictive analysis [13]. Foundation pit deformation is a complex and nonlinear problem. The self-learning and self-adaptive ability of neural networks could be brought into full play. It has its unique advantages in the analysis of foundation pit deformation. BP network is a multilayer feedforward network used in deep foundation pit deformation prediction due to its error backpropagation

characteristics [14, 15]. Because of the deformation's complexity and the limitations of various forecasting models, it is a trend to forecast deformation accurately by using practical information of multiple models [16]. Due to some constraints of the prediction model itself, optimization algorithms or evolutionary algorithms are applied to engineering constructions [17–19].

The GM (1, 1) model and BP neural network were used for in-depth foundation pit prediction research because of their unique advantages. However, in actual application, it can be found that each model also has its shortcomings. This article will introduce a PSO algorithm to improve the two models and compare the prediction accuracy of the improved model and the original model.

2. PSO-GM (1, 1) Model

The establishment of the GM (1, 1) model firstly needs to accumulate the actual sequence once and generate the series $x^{(1)} = (x^{(1)}(1), x^{(1)}(2), \ldots, x^{(1)}(n))$, and use the generated string for analysis and prediction. The newly developed sequence of numbers increases regularly, and then an equation is established for prediction based on the increased regularity.

The time response of the GM (1, 1) model is as follows:

$$\widehat{x}^{(1)}(k+1) = \frac{b}{a} + \left(x^{(0)}(1) - \frac{b}{a} \right) e^{-ak}. \tag{1}$$

It could be seen that the error of the GM (1, 1) model is firstly due to the selection of initial values, and the other is the estimation of gray parameters a and b. $\widehat{u} = \begin{bmatrix} a & b \end{bmatrix}^T = (B^T B)^{-1} B^T Y$

The estimated costs of a and b depend on the construction method of the formula's background value $z^{(1)}(k)$.

Therefore, the GM (1, 1) model's prediction error mainly comes from the limited selection of the initial value and the background value construction formula's error. To reduce the error and improve the model's accuracy, the initial conditions separately and construct the background value was proposed to replace. Reselect the parameters used in the formula to optimize the GM (1, 1) model.

$x1(n)$ could $x^{(1)}(n)$be selected as GM (1, 1)$x^{(1)}(n)x^{(0)}(1)$'s initial condition so that the established model contains the best possible future predictions.

Then the time response sequence of GM (1, 1) is as follows:

$$\widehat{x}^{(1)}(k+1) = \frac{b}{a} + \left(x^{(1)}(n) - \frac{b}{a} \right) e^{-a(k-n+1)}. \tag{2}$$

The simulated value is as follows:

$$\widehat{x}^{(0)}(k+1) = \widehat{x}^{(1)}(k+1) - \widehat{x}^{(1)}(k) \quad k = 1, 2, \ldots, n. \tag{3}$$

When using the GM model for modeling, the construction formula of the background value is as follows:

$$z^{(1)}(k) = 0.5 x^{(1)}(k) + 0.5 x^{(1)}(k-1), \tag{4}$$

If the cumulative sequence slope is slow and the growth trend is not apparent, it is acceptable to use the trapezoidal area to replace the curve's shadow area on the interval. At this time, there is little difference between the two. Still, if the cumulative sequence slope is relatively high, if it is steep and the growth rate is speedy, the trapezoidal area formed is quite different from the interval curve's shadow area. Therefore, the background value construction formula is unreasonable at this time and contains specific errors, as shown in Figure 1.

The background value $z^{(1)}(k)$ is the critical area from the curve in the interval $[k-1, k]$ to the geometric sense's abscissa axis. The proper form of the background value that can satisfy unbiasedness is as follows:

$$z^{(1)}(k) = \int_{k-1}^{k} x^{(1)} dt. \tag{5}$$

According to the integral median theorem, this is also equivalent to the linear combination of $x^{(1)}(k-1)$, $x^{(1)}(k)$ and the parameter λ; that is, there is $\lambda \in [0,1]$ makes the following:

$$z^{(1)}(k) = \lambda x^{(1)}(k-1) + (1-\lambda) x^{(1)}(k). \tag{6}$$

The value of the parameter λ will have a direct impact on the model prediction accuracy. At this time, we have a model parameter optimization problem, combined with the characteristics of the particle swarm algorithm that is good at global optimization, so that complicated formula derivation was avoided, and the thinking is clear and easy to implement.

The following is the process of the PSO algorithm to find the optimal λ:

Step 1: Initialize the population. Randomly generate a sequence on the interval $[0, 1]$:$\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_n)$, each of λ represents the possible weight of the randomly generated background value.

Step 2: Calculate fitness. The fitness function should choose the mean square error function SSE, replace all the generated λ_i to calculate the background value, construct a new background value formula, and use it in the GM (1, 1) model for prediction and the predicted value. The average value of the absolute value of the relative error of the true value is used as the fitness value of the corresponding individual.

Step 3: Compare the current fitness value of the particle with the optimal historical value. If it is better than the optimal historical value, replace its current position with the best position of the particle; compare the current fitness value of the particle with the optimal group value, if it is excellent. If it is at the optimal value of the group, its current position will replace the best position of the group.

Step 4: Update the particles according to the following formula:

$$v_{id}^{k+1} = \omega \cdot v_{id}^k + c_1 \cdot \text{rand}(\,) \cdot \left(p_{id}^k - x_{id}^k \right)$$
$$+ c_2 \cdot \text{rand}(\,) \cdot \left(p_{gd}^k - x_{id}^k \right), \quad (7)$$
$$x_{id}^{k+1} = x_{id}^k + v_{id}^{k+1},$$

In the formula, $c1$ and $c2$ called learning factors or acceleration factors; rand() is a random number between (0, 1); v_{id}^k and x_{id}^k are the velocity of the particle i in the dth dimension in the kth iteration, respectively, And position p_{id}^k is the position of the extreme individual value of the particle i in the dth dimension; p_{gd}^k is the position of the extreme global value of the group in the dth dimension.

Step 5: If the fitness value is not sufficiently good or the preset maximum number of iterations was not obtained, return to step (2).

Through the above steps, the optimal λ_i was found through iteration, and then the optimal background value construction formula is calculated, and then the next step is predicted, as shown in Figure 2.

3. Determination of PSO-BP Network

The combination of BP network and PSO algorithm uses the global search and local search capabilities of both giving full play to their respective advantages, preventing the network from overfitting and falling into local extremums, and at the same time obtaining a faster convergence speed. Randomly initialize the position of the particles in the PSO algorithm. The position of each particle corresponds to a set of weights and thresholds in the BP network. The PSO algorithm was used to iteratively train the network until the optimal particle position is output, the optimal weight, and the threshold. Finally, use this optimized network for prediction. When simulating actual data, to minimize the neural network's average relative error, its fitness function is as follows:

$$E = \frac{1}{m} \sum_{j=1}^{m} \left(d_j - y_j \right)^2. \quad (8)$$

The specific steps of the PSO-BP algorithm were described as follows (see Figure 3):

(1) Initialization. Set the relevant parameters of the PSO-BP network. Determine network structure, including the number of layers of the system and the number of neurons in each layer. Determine the particle group related parameters, including the initial inertia weight w, the learning factors c1 and c2, the maximum number of iterations T, and the number of swarm N. When determining the number of populations, the dimension D of the particles to be optimized should be considered the total number of weight thresholds. The total number of weight thresholds that the PSO algorithm needs to optimize is the following:

$$D = (l + 1)^* n + (n + 1)^* m, \quad (9)$$

l is the number of input neurons; n is the number of hidden layer neurons; m is the number of neurons in the output layer. Finally, the particle's velocity and position were randomly initialized.

(2) Calculating the fitness value of each particle according to the fitness function selected by the problem. The fitness function is as follows:

$$E = \frac{1}{m} \sum_{j=1}^{m} \left(d_j - y_j \right)^2. \quad (10)$$

(3) Compare the current fitness value and the optimal historical value of the particle. If it is better than the optimal historical value, replace its current position with the particle's best position; compare its current fitness value with the optimal amount of the group. For the group's optimal cost, replace its current situation with the best part of the group.

(4) Iterative evolution of the velocity and position of the particle according to the speed and position update formula.

$$v_{id}^{k+1} = \omega \cdot v_{id}^k + c_1 \cdot \text{rand}(\,) \cdot \left(p_{id}^k - x_{id}^k \right)$$
$$+ c_2 \cdot \text{rand}(\,) \cdot \left(p_{gd}^k - x_{id}^k \right), \quad (11)$$
$$x_{id}^{k+1} = x_{id}^k + v_{id}^{k+1}.$$

(5) If the excellent fitness value or the preset maximum iteration number was not reached, then return to step (2). If the condition was met, the global optimal particle position of the output is the optimal BP network weight and threshold.

(6) After the weight threshold is output, the neural network training continued until the result is output.

4. PSO-GM-BP Model Combination Forecast

The foundation pit's deformation process is usually a nonstationary process, which usually presents trend and randomness characteristics. Therefore, the deformation monitoring data could be decomposed into trend items and random items. It is an effective method to use a suitable prediction model to predict each decomposition item accurately (see Figure 4).

The gray forecast model is good at extracting the trend item information contained in the deformation information. It has its unique advantages for forecasting trend items by accumulating and generating new data columns to enhance the trend characteristics. Foundation pit deformation is a complex and nonlinear problem, and the self-learning and self-adaptive capabilities of the BP neural network could be used well. The randomness of deformation could be extracted with high precision. The combined prediction of GM (1, 1) and BP network had been widely used in foundation pit monitoring, but the two models participating in
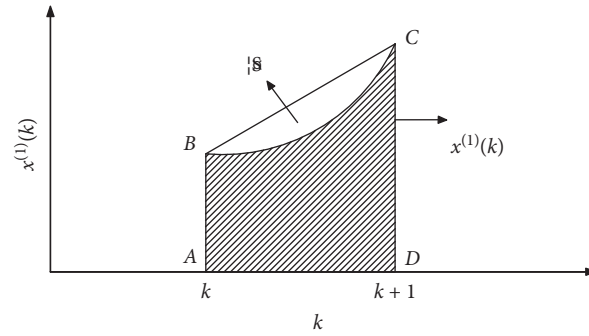
FIGURE 1: Error generating mechanism of background value.
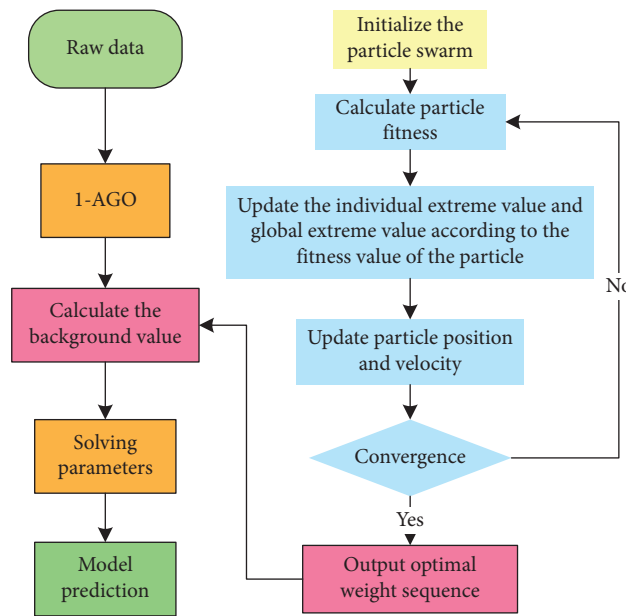
FIGURE 2: Flow chart of particle swarm optimization GM (1, 1) model.

combined forecasts were connected with their shortcomings. They also bring about some errors that could have been avoided. This paper adopts the optimized PSO-BP network and PSO-GM (1, 1) model combination prediction under the premise of small samples, comprehensively considering the advantages of high precision and less information and analyzing the foundation's deformation data pits. Make accurate predictions to guide the safe construction of foundation pits.

The modeling steps of combined forecasting are as follows (see Figure 5):

(1) Use the PSO-GM (1, 1) model to fit and predict the original data, extract the trend items contained therein, and obtain the predicted value and the corresponding residual sequence:

$$\hat{x}^{(0)}(k) = \left\{\hat{x}^{(0)}(1), \hat{x}^{(0)}(2), \ldots, \hat{x}^{(0)}(n)\right\}, \tag{12}$$

$$e(k) = x^{(0)}(k) - \hat{x}^{(0)}(k) = \{e(1), e(2), \ldots, e(n)\}. \tag{13}$$

(2) Use the residual sequence to train the PSO-BP network and obtain the residual after the second correction:

$$\hat{e}(k) = \{\hat{e}(1), \hat{e}(2), \ldots, \hat{e}(n)\}. \tag{14}$$

(3) Finally, add the predicted value of the PSO-GM (1, 1) model and the corrected residual to get the final predicted value:

$$\hat{x}(k) = \hat{x}^{(0)}(k) + \hat{e}(k). \tag{15}$$
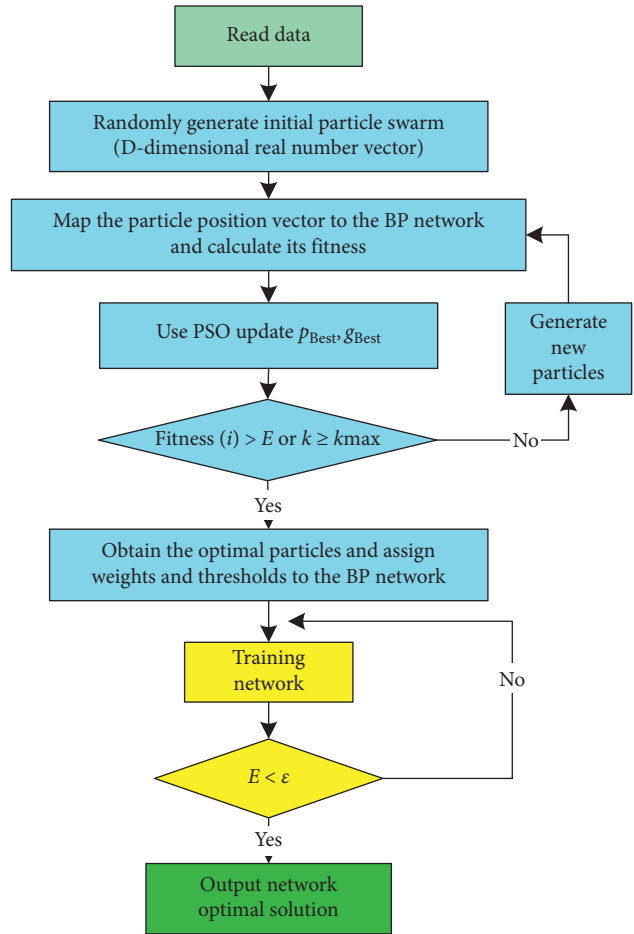
FIGURE 3: PSO- BP network flow chart.
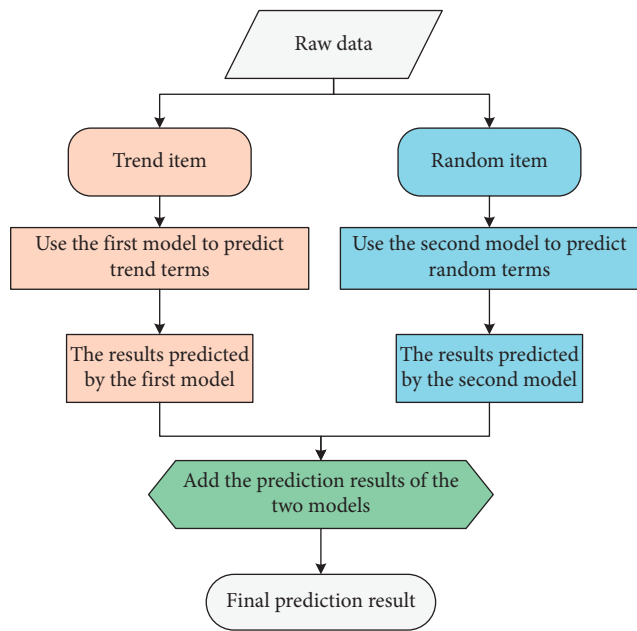
FIGURE 4: Combined forecasting flow chart.
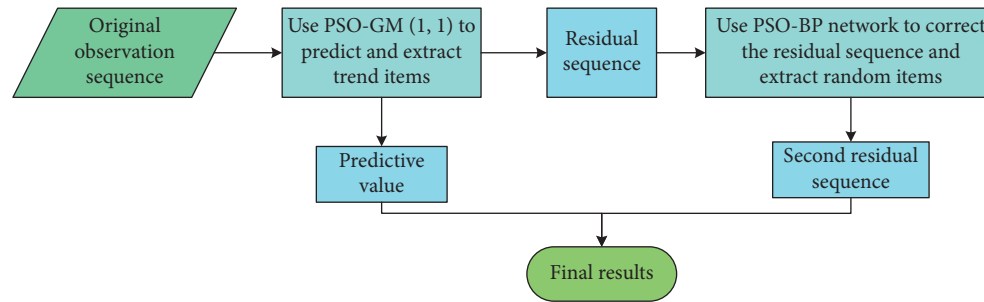
FIGURE 5: PSO-GM (1, 1) and PSO-BP network combination forecasting.

5. Engineering Application Example

5.1. Project Overview. Yingwanzhen Station is the tenth station of the first phase of Changsha Metro Line 4 (Purui Avenue ~ Guihua Avenue), and it also is the interchange station of Line 2 and Line 4. The mileage of the significant platform center of Yingwanzhen Station is DK27 + 192.000, and the platform width is about 14 m. The station is located in the Greenland Group's development zone on the north side of Fenglin 1 road and Lushan Road and passes through Yingwan Road, which was arranged in a north-south direction. Yingwanzhen Station is an island platform station with four underground floors (partial three feet). The total length of outsourced stations is 272.7 m, and the standard section width is 23.3 m. The main body of Yingwanzhen Station was constructed by the open-cut method. The foundation pit's depth is 28.14 m during construction, and the depth of the support structure is up to 36.58 m. The station's enclosure structure to the south of the 13th axis adopts a 1000 mm underground continuous wall with a standard section width of 6 m; the north area adopts the grading excavation + drilling pile support method, with a pile diameter of 1200 mm and a pile spacing of 1350 mm. A row of 34 vertical columns was set along the foundation pit's centerline, which reduces the span and improves the support's bending performance. Two rows of uplift piles were set along the foundation pit's two sides of the centerline. The mounds have a diameter of 1500 m and a total of 66 stacks. The depth can reach 50 m. The shield shaft at the north end of the station uses two concrete supports, and the beam at the south end uses five substantial supports; the standard section uses seven supports for the −25 axis and four supports for the 26 axes to the 34 axes. The first and third supports of the station are concrete supports, and the second, fourth, and fifth supports are steel supports. Both ends of the station are shield starting wells, with four securities starting one after another. The project is large, and the construction period is tight. It is a control site on the whole line.

5.2. Data Selection. Priority should be given to the analysis of the high-risk supporting force and inclinometer data. The subsequent deformation trend could be effectively used to ensure the safety of the foundation pit. The actual measurement data at 15 m (after this referred to as B16) of the

support force monitoring point ZL2-7 in the regular monitoring period and the wall inclinometer monitoring point B16 during the alarm period were selected for analysis and verification. The monitoring data of ZL2-7 and B16 are as follows, as shown in Tables 1 and 2.

As shown in Table 2, the B16 data is more significant than 30 mm from the 11th-period information, which belongs to the data exceeding the alarm value in the deformation monitoring work and belongs to the orange warning category. How to carry out the next construction while ensuring that the monitoring data does not continue to increase is a big problem. It is a more significant challenge for the safety of the entire foundation pit. Therefore, accurate prediction of the data at this point becomes particularly important.

5.3. Support Force Prediction and Analysis

5.3.1. PSO-GM (1, 1) Model Prediction. The GM (1, 1) model and the optimized PSO-GM (1, 1) model were used to model and analyze the measured data. The data used is the 15-period data of ZL2-7 monitoring points. Two models are used to fit and analyze the first 12-period data to predict the 13th to 15th periods' deformation and finally evaluate the two models' accuracy.

The calculation results based on GM (1, 1) model and the PSO-GM (1, 1) model are shown in Table 3, and the accuracy evaluation results are shown in Table 4. The prediction results are based on the original grey model and based on the PSO-GM (1, 1) model shown in Table 5.

The posterior difference ratios based on the original gray model and based on the PSO-GM (1, 1) model are 0.1424 and 0.1269. The small error probability of the two models is both 1. The original gray model's accuracy evaluation results and the PSO-GM (1, 1) model meet the prediction model's excellent standards. Still, the variance based on the PSO-GM (1, 1) model is smaller, indicating that its prediction accuracy is better. It can be seen from Table 5 that for the prediction results, the relative errors of the GM (1, 1) model are 2.67%, 8.31%, and 8.59%, respectively, while the relative errors of the PSO-GM (1, 1) model are 1.69 %, 5.73%, 5.06%. The prediction accuracy based on the PSO-GM (1, 1) model is higher, and the model and prediction results are more stable (see Figure 6). The relative errors and posterior difference ratios were significantly reduced, indicating that the

TABLE 1: ZL2-7 monitoring value.

Period	Monitoring value (kN)
1	1091.74
2	1227.01
3	1343.88
4	1578.08
5	1757.89
6	1822.97
7	1828.44
8	1908.81
9	2061.54
10	2025.6
11	2151.51
12	2221.34
13	2350.88
14	2342.03
15	2455.03

TABLE 2: Cumulative value of B16 monitoring points.

Period	Cumulative value (mm)
1	21.66
2	23.30
3	23.90
4	23.25
5	24.50
6	25.31
7	27.95
8	29.04
9	28.23
10	29.44
11	30.38
12	30.81
13	31.23
14	31.92
15	32.44

accuracy and stability of the model have been greatly improved. The results show that the PSO algorithm is effective and stable for the optimization of the GM (1, 1) model.

5.3.2. PSO-BP Model Prediction. The BP network and the PSO-BP network were used to predict and analyze the monitoring data. The data used is the 15th phase of the ZL2-7 monitoring point. The monitoring data of degrees 1–12 were used as training samples, and the data of the latter three stages are test samples.

To avoid numerical problems and improve network convergence, normalize the sample data before training. The input neuron was set to 5, and the output neuron was 1. The period 1–5 was used as the input, and the data of period six was used as the output. After normalization, the result is output. The number of hidden layer neurons tested to find that the prediction effect is the best when the number of nodes is 6.

The following are the training effect diagrams of the BP network and the PSO-BP system (see Figures 7 and 8):

The target error is achieved in the BP network training diagram when the network training reaches 142 times. In the

PSO-BP network training diagram, the network converges quickly because of the particle swarm algorithm's optimal initial weights and thresholds. The training goal was reached at the 68th training. It shows that the convergence of the BP network optimized by PSO was strengthened.

When using the two systems for prediction, the training effect is very close (see Figures 9 and 10). Still, in terms of generalization ability, as shown in Figures 11 and 12, the PSO-BP system has significantly improved compared to the BP network. The comparison of the overall data is as follows (see Tables 6 and 7, and Figure 13). As shown in Tables 6 and 7, the relative errors of the BP network are 5.21%, 3.07%, and 4.47%, respectively, while the relative errors of the PSO-BP network are 1.95 %, 2.00%, and 2.01%. The relative errors were significantly reduced, indicating that the accuracy and stability of the model have been greatly improved. The results show that the PSO algorithm is effective and stable for the optimization of the BP network.

5.3.3. PSO-GM-BP Model Combination Forecast. It could be seen from Table 6 that the monitoring value of ZL2-7 has an apparent increasing trend, and there is also large volatility. Therefore, the ZL2-7 monitoring value can be decomposed into trend items and random items to establish a combined forecasting model in the integrated forecasting. The data used is the 15-period data of the ZL2-7 monitoring point. The PSO-GM (1, 1) model is used to fit and analyze the first 12 periods' data, and the trend item was predicted for the 13–15 periods. The PSO-BP network is used for residual error correction random item prediction (see Figures 14 and 15).

As shown in Tables 8 and 9, the relative errors of the PSO-GM (1, 1) model were 1.69%, 5.73%, and 5.06%, and the relative errors of the PSO-BP network were 1.95%, 2.00%, and 2.01%, respectively, while the relative errors of the PSO-BP network were 1.45%,0.60%, and 1.94%. The average relative errors of the three models were 4.16%,1.98%, and 1.33%. The accuracy of the three models is sorted from high to low as follows: PSO-GM-BP model, PSO-BP network, and PSO-GM (1, 1) model. With the optimization of the PSO algorithm, the accuracy, efficiency, and stability of the GM (1, 1) model and BP network have been improved. The PSO-GM-BP model has the best prediction effect and can provide effective and efficient data support for foundation pit construction.

5.4. Inclination Data Prediction and Analysis

5.4.1. PSO-GM (1, 1) Model Prediction. The original GM (1, 1) model and the optimized PSO-GM (1, 1) model were used to model and analyze the measured data. The data used is the 15-period data of the B16 monitoring point. The calculation results were shown in Table 10, and the accuracy evaluation results shown in Table 11.

Through the calculation results in Table 11, it could be found that the posterior difference ratios based on the original gray model and the PSO-GM (1, 1) model are 0.1508 and 0.1072, respectively, and the small error probability of

TABLE 3: Actual value and model fitting value of ZL2-7.

Period	Actual value	GM (1, 1)		PSO-GM (1, 1)	
		Fitting value	Relative error (%)	Fitting value	Relative error (%)
1	1091.74	1091.74	0	1022.855	−6.30
2	1227.01	1396.89	−13.85	1397.199	−13.82
3	1343.88	1468.10	−9.24	1469.021	−9.31
4	1578.08	1542.94	−2.23	1544.535	−2.12
5	1757.89	1621.59	−7.75	1623.931	−7.62
6	1822.97	1704.25	−6.51	1707.408	−6.33
7	1828.44	1791.12	−2.04	1795.176	−1.81
8	1908.81	1882.42	−1.38	1887.456	−1.11
9	2061.54	1978.38	−4.03	1984.479	−3.73
10	2025.6	2079.23	−2.65	2086.49	−3.00
11	2151.51	2185.22	−1.57	2193.745	−1.96
12	2221.34	2296.61	−3.39	2306.513	−3.83

TABLE 4: Accuracy assessment table.

Model	C	P
GM (1, 1)	0.1424	1
PSO-GM (1, 1)	0.1269	1

TABLE 5: Comparison of actual and predicted values of ZL2-7.

	Period	13	14	15
	Actual value	2350.88	2342.03	2455.03
GM (1, 1)	Predicted value	2413.68	2536.72	2666.03
	Relative error (%)	−2.67	−8.31	−8.59
PSO-GM (1, 1)	Predicted value	2390.73	2476.32	2579.31
	Relative error (%)	−1.69	−5.73	−5.06

the two models is both 1. The original gray model's accuracy evaluation results and the PSO-GM (1, 1) model meet the prediction model's excellent standards. Still, the variance based on the PSO-GM (1, 1) model is smaller, indicating that its prediction accuracy is better. It can be seen from Table 5 that for the prediction results, the relative errors of the GM (1, 1) model were 3.55%, 3.37%, and 3.62%, respectively, while the relative errors of the PSO-GM (1, 1) model were 2.18 %, 1.98%, and 2.30%. The prediction accuracy based on the PSO-GM (1, 1) model is higher, and the model and prediction results are more stable (see Figure 16). The relative errors and posterior difference ratios were significantly reduced, indicating that the accuracy and stability of the model have been greatly improved. The results show that the PSO algorithm is effective and stable for the optimization of the GM (1, 1) model.

5.4.2. PSO-BP Model Prediction. The BP network and the PSO-BP network were used to predict and analyze the monitoring data. The data used is the 15th phase of the B16 monitoring point. The monitoring data of degrees 1–12 were used as training samples, and the data of the latter three stages are test samples.

To avoid numerical problems and improve network convergence, normalize the sample data before training. After normalization, the result is output. The number of hidden layer neurons was tested repeatedly to find that when the number of nodes is 7, the prediction effect is the best.

The following are the training effect diagrams of the BP network and the PSO-BP system:

In the BP network training diagram, the target error was achieved when the network training reaches 200 times; in the PSO-BP network training diagram, the network performance was degraded because of the particle swarm algorithm's optimal initial weight and threshold. The training goal was reached on the 128th training session. It shows that the convergence of the BP network optimized by PSO was strengthened (see Figures 17 and 18).

When using two networks for prediction, the training data is not much different (see Figures 19 and 20). Still, in terms of generalization ability, as shown in Figures 21 and 22, the PSO-BP system significantly improves the BP network. The comparison of the overall data is as follows (see Tables 12 and 13 and Figure 23). As shown in Tables 12 and 13, the relative errors of the BP network were 9.54%, 6.21%, and 3.93%, respectively, while the relative errors of the PSO-BP network were 0.11%, 0.96%, and 0.82%. The relative errors were significantly reduced, indicating that the accuracy and stability of the model have been greatly improved. The results show that the PSO algorithm is effective and stable for the optimization of the BP network.

5.4.3. PSO-GM-BP Model Combination Forecast. It could be seen that the B16 monitoring value also has an obvious increasing trend and considerable volatility. Therefore, in combined forecasting, the B16 monitoring value can be decomposed into trend items and random items to establish a combined forecasting model. The data used is the 15-period data of the B16 monitoring point. The PSO-GM (1, 1) model is used to fit and analyze the first 12-period data, the trend item was predicted for the 13–15 period, and the PSO-BP network is used for residual. Correct the random thing forecast (see Figures 24 and 25).

As shown in Tables 14 and 15, the relative errors of the PSO-GM (1, 1) model were 2.18%, 1.98%, and 2.30%, and the relative errors of the PSO-BP network were 0.11%, 0.96%, and 0.82%, respectively, while the relative errors of the PSO-BP network were 0.20%, 0.10%, and 0.20%. The average
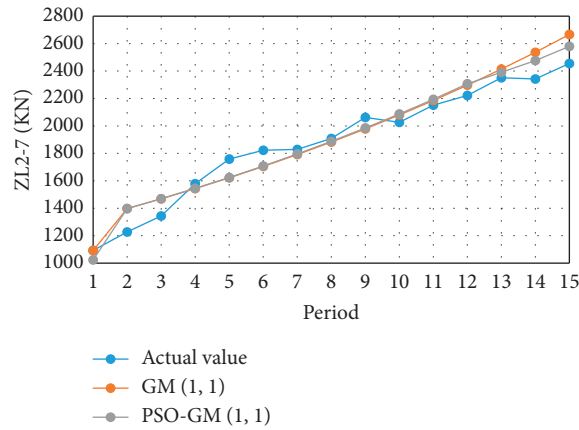
FIGURE 6: Comparison of actual and predicted values.
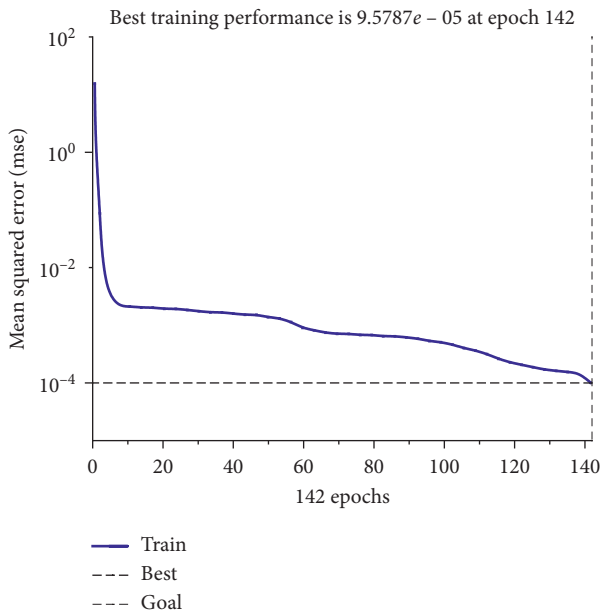
FIGURE 7: BP network training diagram.
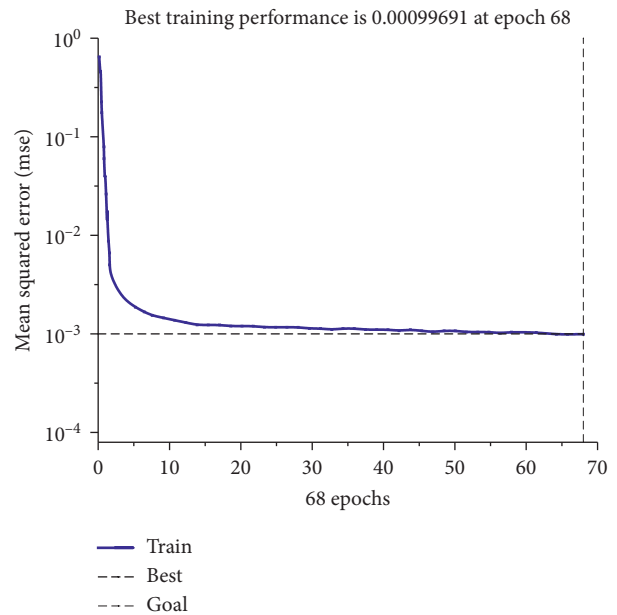
FIGURE 8: PSO-BP network training diagram.

relative errors of the three models were 2.15%, 0.63%, and 0.17%. The accuracy of the three models are sorted from high to low as follows: PSO-GM-BP model, PSO-BP network, PSO-GM (1, 1) model. With the optimization of the PSO algorithm, the accuracy, efficiency, and stability of the GM (1, 1) model and BP network have been improved. The PSO-GM-BP model has the best prediction effect and can provide effective and efficient data support for foundation pit construction.

6. Results and Discussion

(1) The average relative errors of the three models for support force prediction were 4.16%, 1.98%, and 1.33%, and the average relative errors of the three models for support force prediction were 2.15%, 0.63%, and 0.17%.
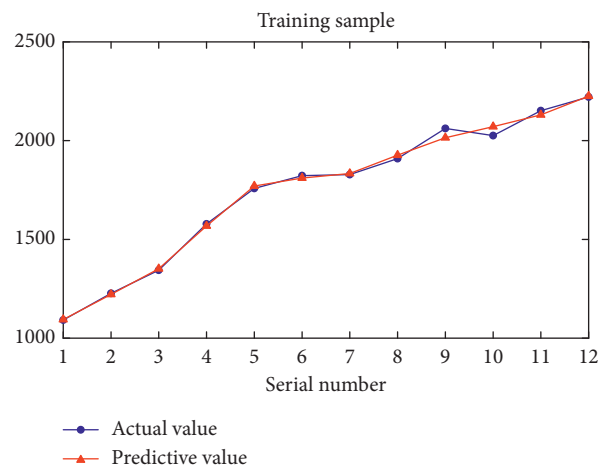
FIGURE 9: BP network training sample.
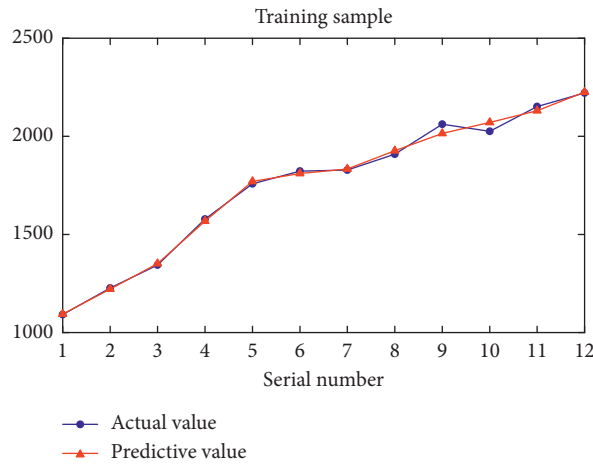
Figure 10: PSO-BP network training sample.
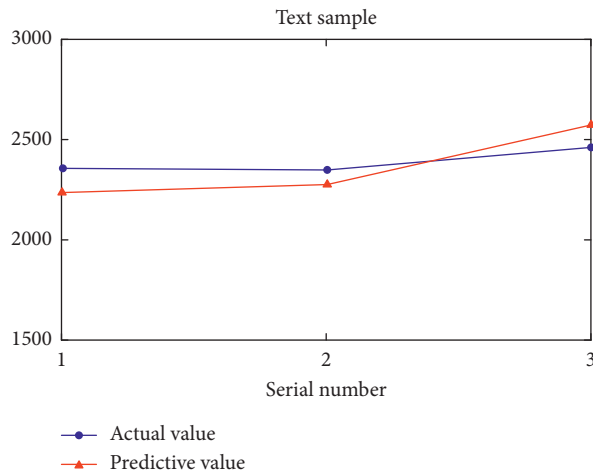
Figure 11: BP network test sample.
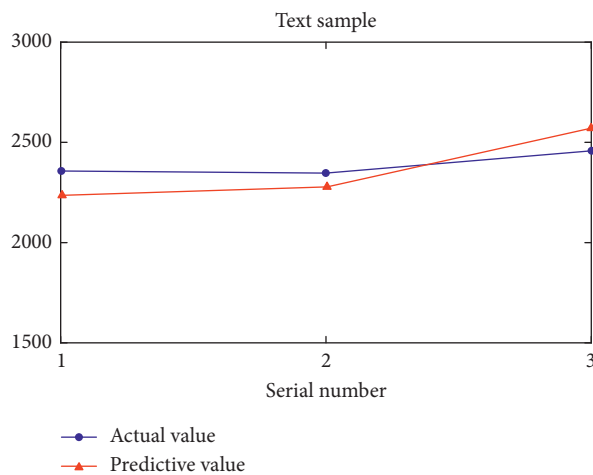
Figure 12: PSO-BP network test sample.

(2) Analyze the error source of the GM (1, 1) model, apply the global optimization feature of the PSO algorithm to optimize the parameters of the background value construction formula of the GM (1, 1) model, and propose an optimized PSO-GM (1, 1) Model. Combining axial force and inclinometer

Table 6: ZL2-7 actual value and model fitting value.

Period	Actual value	BP network		PSO-BP network	
		Fitting value	Relative error (%)	Fitting value	Relative error (%)
1	1091.74	1094.02	−0.21	1096.139	−0.40
2	1227.01	1221.387	0.46	1226.61	0.03
3	1343.88	1351.156	−0.54	1345.639	−0.13
4	1578.08	1567.732	0.65	1576.096	0.12
5	1757.89	1770.027	−0.69	1760.421	−0.14
6	1822.97	1810.918	0.66	1821.395	0.08
7	1828.44	1833.811	−0.29	1828.591	−0.01
8	1908.81	1926.841	−0.94	1908.798	0
9	2061.54	2014.674	2.27	2061.533	0
10	2025.6	2070.983	−2.24	2025.602	0
11	2151.51	2130.434	0.98	2151.51	0
12	2221.34	2225.197	−0.17	2221.34	0

Table 7: ZL2-7 actual value and predicted value comparison results table.

Period		13	14	15
Actual value		2350.88	2342.03	2455.03
BP network	Predicted value	2228.461	2270.236	2564.764
	Relative error (%)	5.21	3.07	−4.47
PSO-BP network	Predicted value	2305.01	2388.83	2405.64
	Relative error (%)	1.95	−2.00	2.01
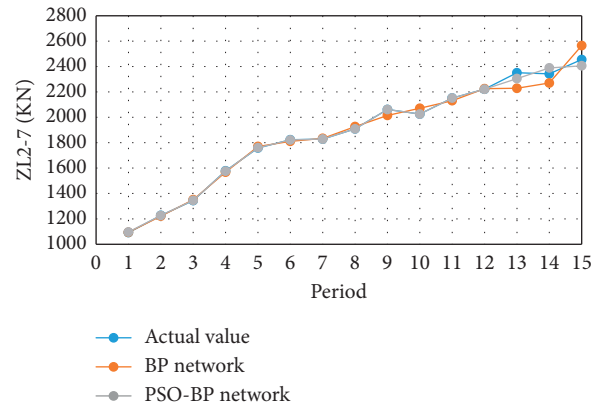
Figure 13: Comparison of actual and predicted values.
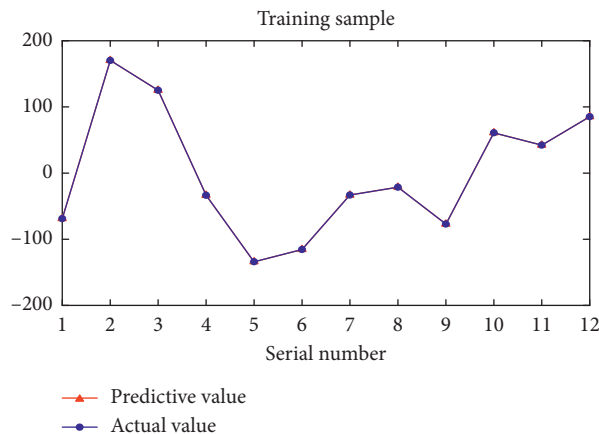
Figure 14: Residual training samples.
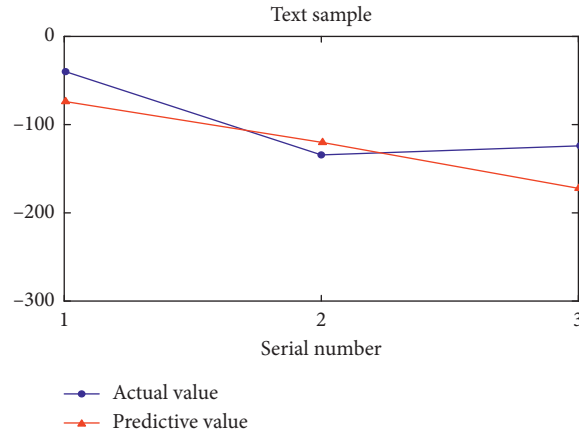
Figure 15: Residual test samples.

Table 8: Comparison of three models.

Period	Actual value	Predicted value		
		PSO-GM (1, 1)	PSO-BP	PSO-GM-BP
13	2350.88	2390.73	2305.01	2316.65
14	2342.03	2476.32	2388.83	2356.05
15	2455.03	2579.31	2405.64	2407.19

Table 9: Comparison of relative error of three models.

Period	Relative error (%)		
	PSO-GM (1, 1)	PSO-BP	PSO-GM-BP
13	−1.69	1.95	1.45
14	−5.73	−2	−0.60
15	−5.06	2.01	1.94
Average relative error	4.16	1.98	1.33

Table 10: Actual and simulated results of B16.

Period	Actual value	GM (1, 1)		PSO-GM (1, 1)	
		Fitting value	Relative error (%)	Fitting value	Relative error (%)
1	21.66	21.6600	0.00	21.1069	2.55
2	23.30	22.8930	1.75	22.6821	2.65
3	23.90	23.6234	1.16	23.4549	1.86
4	23.25	24.3771	4.85	24.2521	4.32
5	24.50	25.1548	2.67	25.0804	2.37
6	25.31	25.9574	2.56	25.9350	2.47
7	27.95	26.7856	4.17	26.8186	4.05
8	29.04	27.6402	4.82	27.7323	4.50
9	28.23	28.5220	1.03	28.6772	1.58
10	29.44	29.4320	0.03	29.6543	0.73
11	30.38	30.3711	0.03	30.6646	0.94
12	30.81	31.3400	1.72	31.7094	2.92
13	31.23	32.3399	3.55	31.9099	2.18
14	31.92	32.9946	3.37	32.5515	1.98
15	32.44	33.6142	3.62	33.1876	2.3

Table 11: Accuracy assessment table.

Model	C	P
GM (1, 1)	0.1508	1
PSO-GM (1, 1)	0.1072	1
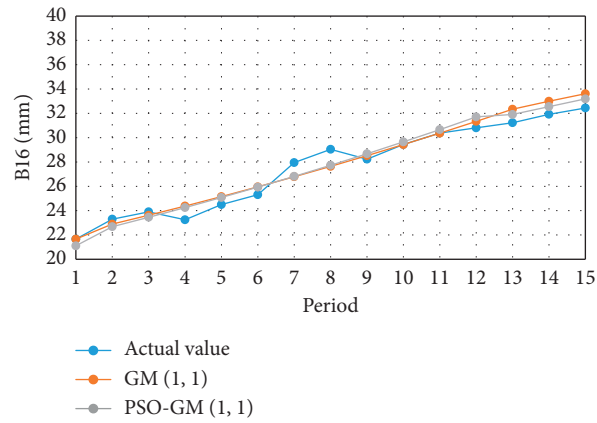
Figure 16: Comparison of actual and predicted values.
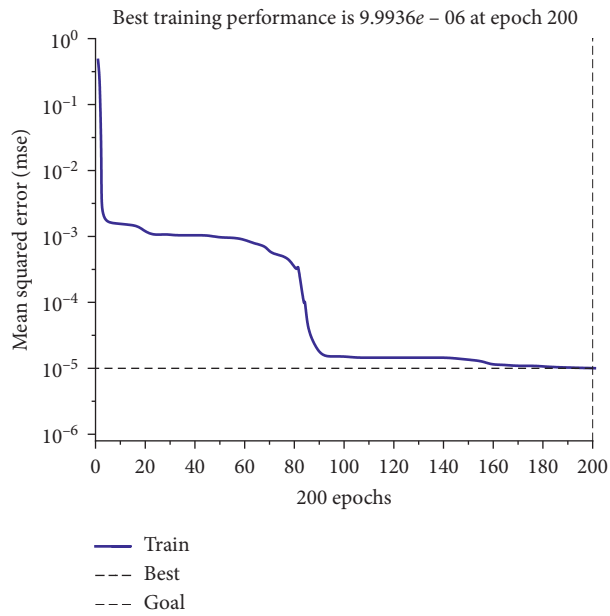
Figure 17: BP network training diagram.
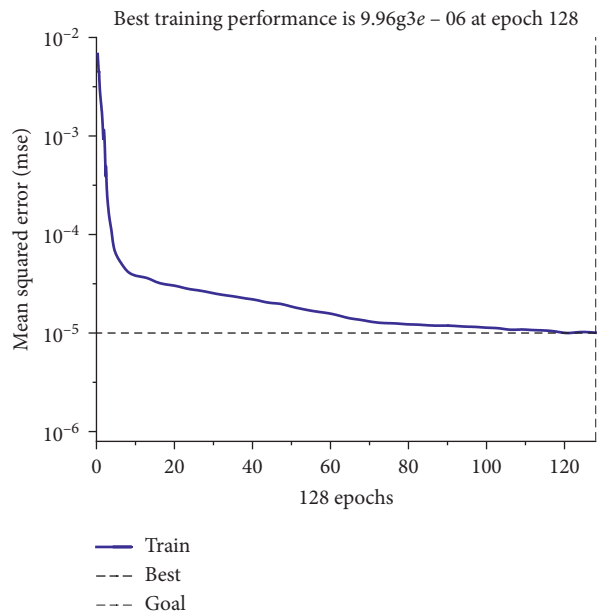
Figure 18: PSO-BP network training diagram.
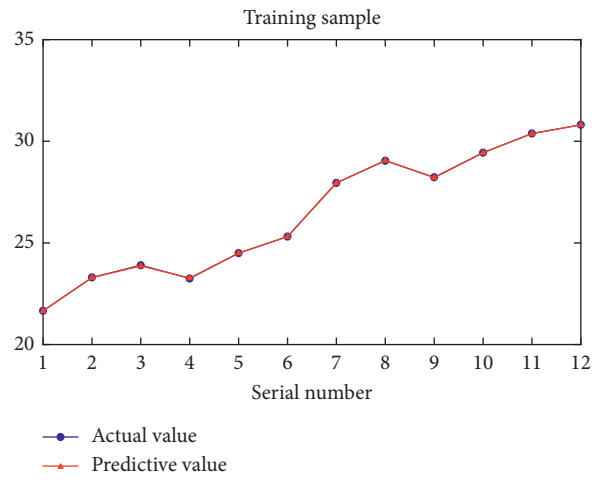
Figure 19: BP network training sample.
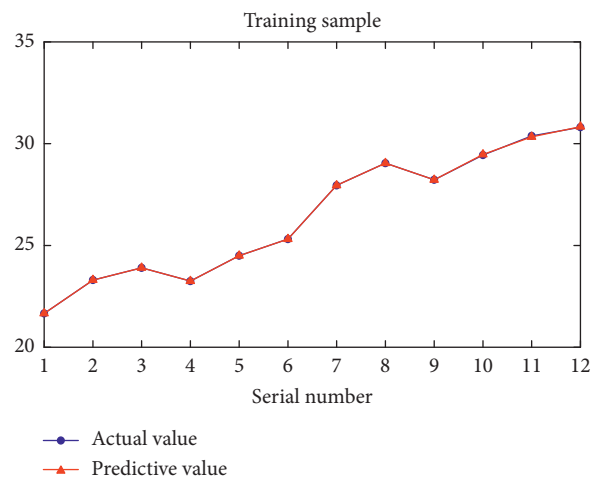
Figure 20: PSO-BP network training sample.
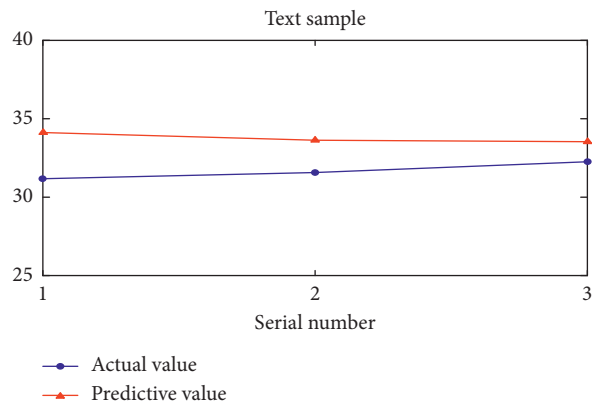
Figure 21: BP network test sample.
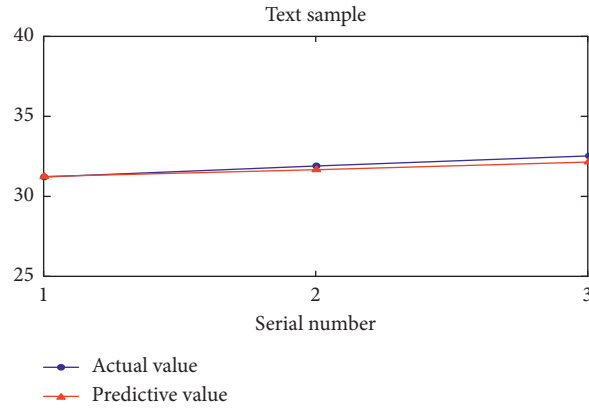
Figure 22: PSO-BP network test sample.

Table 12: B16 actual value and model fitting value results.

Period	Actual value	BP network		PSO-BP network	
		Fitting value	Relative error (%)	Fitting value	Relative error (%)
1	21.66	21.6528	−0.03	21.6594	0.00
2	23.30	23.2941	−0.02	23.2997	0.00
3	23.90	23.8606	−0.16	23.8993	0.00
4	23.25	23.2772	0.12	23.2501	0.00
5	24.50	24.4945	−0.02	24.4982	−0.01
6	25.31	25.3054	−0.02	25.3095	0.00
7	27.95	27.9457	−0.01	27.9483	−0.01
8	29.04	29.0528	0.04	29.0400	0.00
9	28.23	28.2108	−0.07	28.2220	−0.03
10	29.44	29.4375	−0.01	29.4683	0.10
11	30.38	30.3777	−0.01	30.3346	−0.15
12	30.81	30.8088	−0.00	30.8323	0.07

Table 13: B16 actual value and predicted value comparison results.

Period		13	14	15
Actual value		31.23	31.92	32.44
BP network	Predicted value	34.2091	33.9029	33.7154
	Relative error (%)	9.54	6.21	3.93
PSO-BP network	Predicted value	31.1965	31.6132	32.1754
	Relative error (%)	−0.11	−0.96	−0.82
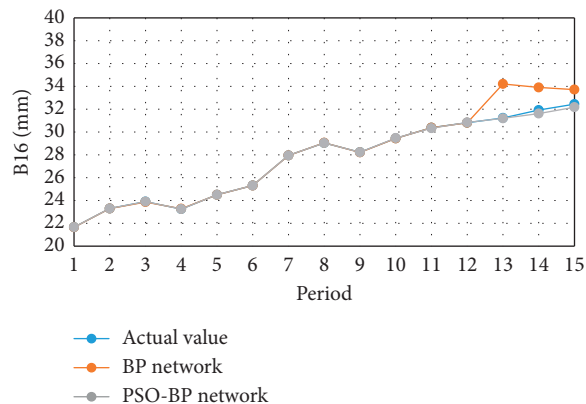
Figure 23: Comparison of the actual and predicted value.
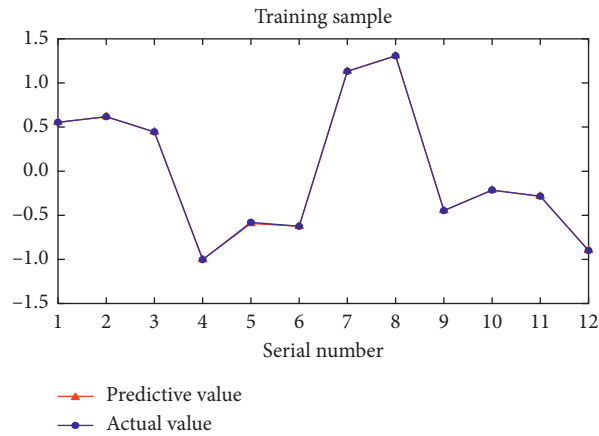
Figure 24: Residual training samples.
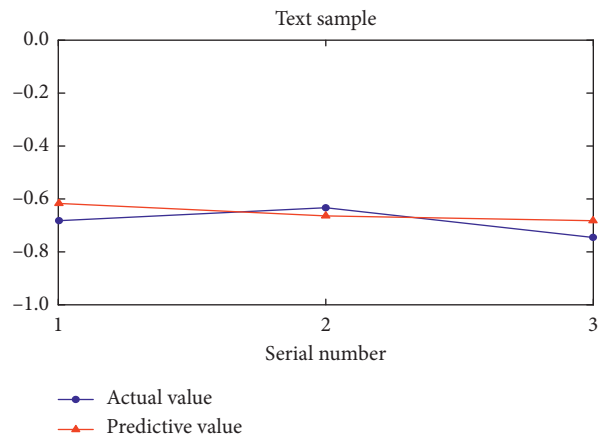
Figure 25: Residual test sample.

Table 14: Comparison of three models.

Period	Actual value	Predicted value		
		PSO-GM (1, 1)	PSO-BP	PSO-GM-BP
13	31.23	31.9099	31.1965	31.2924
14	31.92	32.5515	31.6132	31.8863
15	32.44	33.1876	32.1754	32.5039

Table 15: Comparison of relative error of three models.

Period	Relative error (%)		
	PSO-GM (1, 1)	PSO-BP	PSO-GM-BP
13	−2.18	0.11	−0.20
14	−1.98	0.96	0.10
15	−2.30	0.82	−0.20
Average relative error	2.15	0.63	0.17

data, using GM (1, 1) model and PSO-GM (1, 1) model, respectively, for prediction, PSO-GM (1, 1) model has improved the prediction accuracy of the two kinds of data, and Model stability is also better.

(3) The PSO optimization algorithm is selected to optimize the weight and threshold, and the PSO-BP network model was proposed. Combining the axial support force and inclinometer data obtained on site,

respectively, use the BP network and PSO-BP network to predict. PSO-BP network has significantly improved the prediction accuracy of the two kinds of data, and the convergence speed is faster.

7. Conclusion

(1) Combine the advantages and disadvantages of the PSO-GM (1, 1) model and the PSO-BP network to construct a PSO-GM-BP model. Through the comparative analysis of the prediction results of several models in this article, it could be concluded that the GM (1, 1) model and the BP neural network model can both predict more accurate results. With higher accuracy and more substantial applicability, the PSO-GM-BP model has the best predictive effect and can effectively guide the construction of foundation pit projects.

(2) The PSO algorithm is used to optimize the GM (1, 1) model and the BP network, which strengthens the accuracy and applicability of the model, but at the same time, there are some shortcomings and areas for improvement. While continuing to study, we will also consider a more in-depth comparison with other methods to find a more reasonable solution suitable for engineering intelligent applications.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

References

[1] J. D. Watson and F. H. C. Crick, "Molecular structure of nucleic acids: a structure for deoxyribose nucleic acid," *Nature*, vol. 171, no. 4356, pp. 737–738, 1953.

[2] S.-L. Shen, H.-N. Wu, Y. -J. Cui, and Z.-Y. Yin, "Long-term settlement behaviour of metro tunnels in the soft deposits of shanghai," *Tunnelling and Underground Space Technology*, vol. 40, pp. 309–323, 2014.

[3] Y. Tan and D. Wang, "Characteristics of a large-scale deep foundation pit excavated by the central-island technique in shanghai soft clay. I: bottom-up construction of the central cylindrical shaft," *Journal of Geotechnical and Geoenvironmental Engineering*, vol. 139, no. 11, pp. 1875–1893, 2013.

[4] Z.-J. Luo, Y.-Y. Zhang, and Y.-X. Wu, "Finite element numerical simulation of three-dimensional seepage control for deep foundation pit dewatering," *Journal of Hydrodynamics*, vol. 20, no. 5, pp. 596–602, 2008.

[5] S.-X. He, "Experimental researches on unloading deformation of clay in excavation of foundation pit," *Rock and Soil Mechanics-Wuhan-24.1: ISSU*, vol. 82, pp. 17–20, 2003.

[6] H.-F. Liu, "Study of a gray genetic BP neural network model in fault monitoring and a diagnosis system for dam safety," *ISPRS International Journal of Geo-Information*, vol. 7, no. 1, 4 pages, 2018.

[7] T. Feng, "Research of combination forecasting model based on improved analytic hierarchy process," *EURASIP Journal on Wireless Communications and Networking*, vol. 1, p. 182, 2018.

[8] C. Zhang, J.-Z. Li, and H. E. Yong, "Application of optimized grey discrete verhulst–BP neural network model in settlement prediction of foundation pit," *Environmental Earth Sciences*, vol. 78, no. 15, p. 441, 2019.

[9] W. Zhang, C. Wu, Y. Li, L. Wang, and P. Samui, "Assessment of pile drivability using random forest regression and multivariate adaptive regression splines," *Georisk: Assessment and Management of Risk for Engineered Systems and Geohazards*, vol. 10, pp. 1–14, 2019.

[10] Z. Ji, B. Wang, S. Deng, and Z. You, "Predicting dynamic deformation of retaining structure by LSSVR-based time series method," *Neurocomputing*, vol. 137, pp. 165–172, 2014.

[11] Y. Yao, J. M. Becker, M. R. Ford, and M. A. Merrifield, "Modeling wave processes over fringing reefs with an excavation pit," *Coastal Engineering*, vol. 109, pp. 9–19, 2016.

[12] J. Wang, "Areal subsidence under pumping well–curtain interaction in subway foundation pit dewatering: conceptual model and numerical simulations," *Environmental Earth Sciences*, vol. 75, no. 3, p. 198, 2016.

[13] H. Wu, Y. Dong, W. Shi et al., "An improved fractal prediction model for forecasting mine slope deformation using GM (1, 1)," *Structural Health Monitoring: An International Journal*, vol. 14, no. 5, pp. 502–512, 2015.

[14] W. Xingke and W. Juan, "Study of deformation prediction of foundation pit based on Optimized support vector machine and chaotic BP neural network," *Tunnel Construction*, vol. 37, no. 9, 1105 pages, 2017.

[15] J. Cao, "Time series forecast of foundation pit deformation based on BP neural network," *Applied Mechanics and Materials*, vol. 55, 61 page, 2014.

[16] X. Li, J. Kong, and C. Wang, "Application of combined model with optimum weight in prediction of landslide deformation," *Nature Reviews Disease*, vol. 02, pp. 53–57, 2008.

[17] W. Deng, J. Xu, H. Zhao, and Y. Song, "A novel gate resource allocation method using improved PSO-based QEA," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 5, 9 pages, 2020.

[18] W. Deng, J. Xu, Y. Song, and H. Zhao, "Differential evolution algorithm with wavelet basis function and optimal mutation strategy for complex optimization problem," *Applied Soft Computing*, vol. 2020, Article ID 106724, 2020.

[19] W. Deng, H. Liu, J. Xu, H. Zhao, and Y. Song, "An improved quantum-inspired differential evolution algorithm for deep belief network," *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 10, pp. 7319–7327, 2020.

Research Article

Image Processing-Based Spall Object Detection Using Gabor Filter, Texture Analysis, and Adaptive Moment Estimation (Adam) Optimized Logistic Regression Models

Nhat-Duc Hoang [ID] [1,2]

^{1}*Institute of Research and Development, Duy Tan University, Da Nang 550000, Vietnam*
^{2}*Faculty of Civil Engineering, Duy Tan University, Da Nang 550000, Vietnam*

Correspondence should be addressed to Nhat-Duc Hoang; hoangnhatduc@duytan.edu.vn

This study aims at proposing a computer vision model for automatic recognition of localized spall objects appearing on surfaces of reinforced concrete elements. The new model is an integration of image processing techniques and machine learning approaches. The Gabor filter supported by principal component analysis and k-means clustering is used for identifying the region of interest within an image sample. The binary gradient contour, gray level co-occurrence matrix, and color channels' statistical measurements are employed to compute the texture of the extracted region of interest. Based on the computed texture-based features, the logistic regression model trained by the state-of-the-art adaptive moment estimation (Adam) is utilized to establish a decision boundary that delivers predictions on the status of "nonlocalized spall" and "localized spall." Experimental results demonstrate that the newly developed model is able to achieve good detection accuracy with classification accuracy rate = 85.32%, precision = 0.86, recall = 0.79, negative predictive value = 0.85, and $F1$ score = 0.82. Thus, the proposed computer vision model can be helpful to assist decision makers in the task of the periodic survey of structure heath condition.

1. Introduction

Public safety is a major concern of civil engineers who design and maintain high-rise buildings. Despite considerable efforts in design and advanced knowledge of building structures, accidents can still happen in the built environment due to excessive usage, structural aging, and inclement weather conditions [1]. Among the hazards occurred in high-rise buildings, falling objects from overhead caused by concrete spalling can be particularly dangerous and have a high potential severity to occupants' heath [2]. The effect of concrete debris can be devastating for human lives if it gets broken off from surfaces of exterior wall systems of high-rise buildings [3].

A concrete spall (Figure 1) is regarded as flakes of concrete/mortar broken off from a concrete element (e.g., beam, wall, and ceiling) [4]. Spalling is typically caused by stresses brought about by differential movement of materials. Most often, spalling in concrete is due to corrosion of steel reinforcement embedded in the structure. To prevent such accidents and to ensure the safety and serviceability of the built environment, periodic visual surveys of structural heath condition and proper maintenance activities are very crucial [5].

In developing countries, including Vietnam, manual visual inspection is still the principal method for evaluating structural heath conditions. This activity is performed at regular intervals to identify potential damages and guarantee the service/safety requirements of high-rise buildings. Provided the well-trained technicians experienced in structural heath assessment, manual visual inspection is able to providing accurate surveying outcomes. Nevertheless, due to the increasing numbers of buildings needed to be inspected periodically and the limited number of experience technicians, timely evaluation of building elements becomes infeasible and inspection deficiencies become a major

| (a) | (b) | (c) |
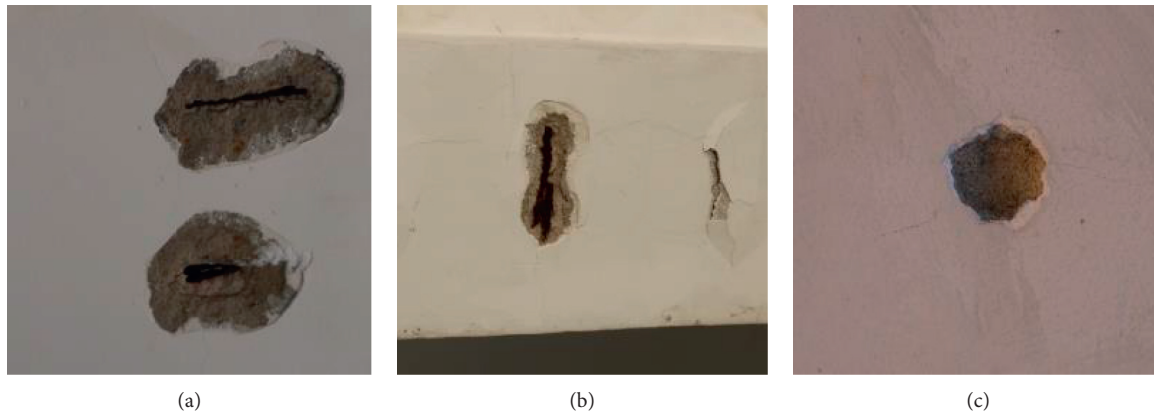
FIGURE 1: Spall objects on (a) ceiling, (b) beam, and (c) wall structures.

concern of property owners. Therefore, there is a practical need to substitute the unproductive manual visual survey with a more effective approach.

Recently, due to the ease of access to low-cost and high-quality visual sensing equipment including digital cameras, computer vision-based models have been increasingly used for automatic structural heath monitoring [6]. These advanced approaches have been proved to be viable alternatives to replace the labor-intensive and subjective methods relied on manual survey. With the use of advanced image processing techniques operated on image samples collected from digital cameras, the physical condition of civil structures can be continuously surveyed and reported to maintenance agencies. This evaluation outcome can be effectively used to support the decision-making process regarding maintenance prioritization and funding allocation.

Due to such reasons, a large number of computer vision-based approaches have been proposed to successfully detect various forms of structural defects such as cracking and spalling. Abdel-Qader et al. [7] employs a principle component analysis-based model to recognize cracking defects appeared on bridge surfaces; the principle component analysis is utilized to support data cluster identification with a large database of bridge images. O'Byrne et al. [8] utilizes texture analysis for detecting damages appeared on infrastructural elements; the texture-based image segmentation relies on pixel intensity values and gray level co-occurrence matrix. Subsequently, a support vector machines model is then employed for the data classification task. Lattanzi and Miller [9] rely on the data clustering approach for image segmentation based on the Canny and k-means algorithm; the research finds that the combined algorithms can deliver good accuracy of crack recognition under different environmental circumstances.

As can be seen from the literature, a large number of previous studies have been dedicated to crack detection for concrete structures [10–20]. Only recently, there is an increasing focus on detecting other forms of damage including concrete spalling [21–24]. German et al. [25] constructs a combination of segmentation, template matching, and morphological preprocessing for detecting spall appeared on surfaces of concrete columns. Machine learning models

including support vector machines, Naïve Bayesian classifier, and random forest have been employed to identify concrete defects [8, 26]. A model for localization and quantification of concrete spalling defects based on terrestrial laser scanning has been proposed in [27]. Dawood et al. [21] presented a computer vision-based model for spalling detection used in environment of subway networks.

Hoang [28] relies on a steerable filter and machine learning to recognize wall defects such as cracks and spalls. A concrete spalling detection model for metro tunnel from point cloud that employs a roughness descriptor has been developed by Wu et al. [24]. Hoang [29] presents an image processing approach for automatic detection of concrete spalling using machine learning algorithms and image texture analysis. Nevertheless, this model focused on machine learning-based texture discrimination and was not capable of isolating the entire individual spall object.

Yao et al. [30] establishes a convolutional neural network-based model for concrete bughole detection; a large number (about 10,000) of image examples have been used as a training dataset. Li et al. [31] proposed a model for detecting exposed aggregate appeared on stilling basin slab using the attention U-Net network. Chow et al. [32] employs deep learning of a convolutional autoencoder for anomaly detection of defects existing on concrete structures. A model for recognizing damaged ceiling areas in large-span structures has been proposed by Wang et al. [33]; this model employs a convolutional neural network for pattern recognition. Although deep learning-based models are capable of performing feature extraction phase automatically, these supervised learning models generally demand a large-size training dataset and a meticulous process of data labeling [34–36]. This data labeling process itself can be time-consuming as well as error prone [5]. In addition, the deep learning models also require experience and the trial-and-error process to adjust a significant amount of model's free parameters.

An effort of combining unsupervised learning and machine learning-based data classification has been recently introduced in [37]. The k-means clustering algorithm and machine learning classifier have been integrated to form an automatic approach for estimating stripping of asphalt

coating. The k-means clustering algorithm is utilized to separate pixels with similar values on the surface of aggregates; subsequently, machine learning models are used to categorize the identified clusters into groups of asphalt-coated and uncoated areas.

As pointed out by previous studies, the current challenges faced by computer vision-based concrete damage detection including spall recognition are complex environmental conditions (e.g., noisy background image) [5] and the difficulty of the image labeling process [32]. More efforts should be dedicated to automatically identify the damage's region of interest (ROI) via unsupervised learning methods. Capable machine learning methods with few free parameters should be investigated as viable alternatives to sophisticated models used for data classification. It is because simple and manageable models significantly facilitate the development and application of hybrid computer vision-machine learning approaches for concrete spalling detection.

Based on such motivation, this study proposes and verifies an automated method for recognizing localized spall objects based on an integration of a Gabor filter, k-means clustering, image texture analysis, and logistic regression pattern classification models. The Gabor filter coupled with the principal component analysis and the k-means clustering are used synergistically for automatic identification of ROI on concrete surface. The image texture analysis combines powerful texture discriminators of binary gradient contours, color channels' properties, and the gray level co-occurrence matrix. The logistic regression model trained by the state-of-the-art adaptive moment estimation (Adam) optimizer is employed for data classification.

The subsequent sections of the study are organized as follows: the second section reviews the research methodology. The third section presents the image data collection process. The proposed integrated model used for concrete spall detection is described in the next section, followed by the experimental results and discussion. The final section summarizes the research findings with several concluding remarks.

2. Research Methodology

2.1. Gabor Filter (GF).
Image segmentation is the process of separating an image into distinctive regions [38, 39]. The GF is a widely applied approach for segmenting image [40, 41]. This approach is inspired by the multichannel operation of the human visual system used for visual interpretation in real-world circumstances [42–44]. Based on experimentation, it has been shown that the GF resembles simple cells in the Mammalian vision system. Thus, this filter can be a reasonable model of how humans actually recognize and discriminate areas characterized by different texture [45].

The GF consists of two-dimensional Gabor filters which can be described as complex sinusoidal waves modulated by Gaussian envelopes [43]. This filter carries out a localized and oriented frequency analysis of a two-dimensional signal. The GF yields a response that can be mathematically given as follows [45]:

$$h(x, y) = \exp\left\{-\frac{1}{2}\left[\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}\right]\right\}\cos(2\pi u_0 x), \qquad (1)$$

where u_0 denotes the frequency of a sinusoidal plane wave along the x axis. σ_x and σ_y represent the space constants of the Gaussian envelope along the x and y axes, respectively. It is noted that the GF with arbitrary orientations can be obtained via a rigid rotation of the x-y coordinate system.

The frequency domain representation of the GF is given by [45]

$$H(u, v) = A\left(\exp\left\{-\frac{1}{2}\left[\frac{(u - u_0)^2}{\sigma_u^2} + \frac{v^2}{\sigma_v^2}\right]\right\}\right.$$
$$\left. + \exp\left\{-\frac{1}{2}\left[\frac{(u + u_0)^2}{\sigma_u^2} + \frac{v^2}{\sigma_v^2}\right]\right\}\right), \qquad (2)$$

where $\sigma_u = 1/2\pi\sigma_x$, $\sigma_v = 1/2\pi\sigma_y$, and $A = 2\pi\sigma_x\sigma_y$.

It is worth noticing that it is necessary to specify tuning parameters of the GF including the orientation angles and the radial frequency. Based on the suggestions of Jain and Farrokhnia [45], four values of orientations are often employed: $0°$, $45°$, $90°$, and $135°$. Given an image with a width of N_w pixels, the following values of radial frequency u_0 can be considered: $1\sqrt{2}, 2\sqrt{2}, 4\sqrt{2}, ..., N_w/4\sqrt{2}$.

2.2. The K-Means Clustering Algorithm.
In this study, the unsupervised machine learning approach of k-means clustering [46] is employed to divide an image into different regions based on the analysis results obtained from the GF. This unsupervised machine learning method is simple yet powerful algorithm for automatic data grouping [47]. Based on such method, image pixels that have the similar properties can be grouped in one cluster. Accordingly, data samples belonging to one cluster feature the smallest degree of variation. The iterative algorithm used to compute the cluster centers is presented in Algorithm 1.

2.3. Image Texture Analysis

2.3.1. Binary Gradient Contours (BGC).
The BGC, proposed by Fernández et al. [48], is a group of computationally simple texture descriptors. Given a 3×3 grayscale image patch, these texture descriptors employs a set of eight binary gradients between pairs of pixels all along a closed path around the central pixel [49]. The BGC includes three versions which are single-loop, double-loop, and triple-loop descriptors. Via experimentation, the BGC operator has been found to achieve good texture discrimination outcomes.

A matrix S which is the pixel intensity of an image patch of the size 3×3 is given as follows:

$$S = \begin{bmatrix} I_7 & I_6 & I_5 \\ I_0 & I_c & I_4 \\ I_1 & I_2 & I_3 \end{bmatrix}, \qquad (3)$$

```
      Determine the number of cluster k
      Randomly assign k centers of data samples
(1) Loop
(2)       Assign each data points to the cluster with the nearest mean
(3)       Recalculate means for data points assigned to each cluster
(4) Until the data assignments are unchanged.
```
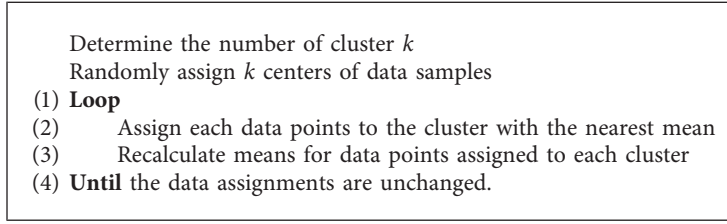
ALGORITHM 1: The k-means clustering.

where I_c denotes the central pixel. I_0, I_1, ..., I_7 are the neighboring pixels.

The schematic representations of BGC with three versions of single, double, and triple loops are presented in Figure 2. In addition, to facilitate the mathematical formulation of these texture descriptors, a square crop $S_{m,n}$ is given by

$$S = \begin{bmatrix} I_{m-1,n-1} & I_{m-1,n} & I_{m-1,n-1} \\ I_{m,n-1} & I_{m,n} & I_{m,n+1} \\ I_{m+1,n-1} & I_{m+1,n} & I_{m+1,n+1} \end{bmatrix}, \quad (4)$$

where $I_{m,n}$ represents the pixel at m^{th} row and n^{th} column.

Accordingly, the formulations of the single, double, and triple-loop versions are given by [48]

(i) Single-loop version:

$$\text{BGC}_1 = \sum_{n=0}^{7} \lambda \left(I_n - I_{(n+1) \bmod 8} \right) \times 2^n - 1. \quad (5)$$

(ii) Double-loop version:

$$\text{BGC}_2 = 15 \times \sum_{n=0}^{3} \lambda \left(I_{2n} - I_{2(n+1) \bmod 8} \right) \times 2^n$$
$$+ \sum_{n=0}^{3} \lambda \left(I_{2n+1} - I_{(2n+3) \bmod 8} \right) \times 2^n - 16. \quad (6)$$

(iii) Triple-loop version:

$$\text{BGC}_3 = \sum_{n=0}^{7} \lambda \left(I_{3n \bmod 8} - I_{3(n+1) \bmod 8} \right) \times 2^n - 1,$$
$$\text{where } \lambda(x) = \begin{cases} 1, & if \ x \geq 0, \\ 0, & if \ x < 0. \end{cases} \quad (7)$$

2.3.2. RGB Channels' Properties.
Since the color properties of spall and nonspall objects are expected to be dissimilar, this study employs the statistical measurements of three color channels: red (R), green (G), and blue (B) as a means of texture description. Given an image sample I, the first-order histogram $P(I)$ can be computed. Accordingly, the mean (μ_c), standard deviation (σ_c), skewness (S_c), kurtosis (K_c), entropy (E_c), and range (R_c) of the three color channels (R, G, and B) can be calculated [29, 50].
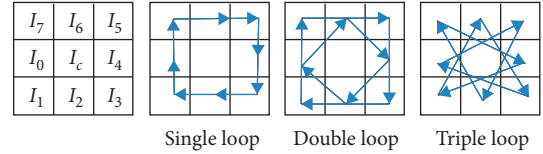
FIGURE 2: The graphical representation of BGC.

2.3.3. Gray Level Co-Occurrence Matrix (GLCM).
The GLCM [51, 52] is also an extensively employed approach for characterizing image texture. This approach focuses on capturing the repeated occurrence of certain gray-level patterns [53]. Therefore, indices extracted from a GLCM can be effectively utilized to evaluate the coarseness/fineness of an image region. Let r and θ represent a distance and a rotation relationship between two individual pixels. The GLCM, denoted as P_δ, denotes a probability of the two gray levels of i and j having the relationship specifying by r and θ [54]. Based on the recommendations of Haralick et al. [51], the GLCM can be constructed with $r = 1$ and $\theta = 0°$, $45°$, $90°$, and $135°$. Accordingly, for each matrix, four indices of angular second moment (AM), contrast (CO), correlation (CR), and entropy (ET) can be computed as follows [29, 55]:

$$\text{AM} = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} P_\delta^N (i, j)^2,$$

$$\text{CO} = \sum_{k=0}^{N_g-1} k^2 \sum_{\substack{i=1 \\ |i-j|=k}}^{N_g} \sum_{j=1}^{N_g} P_\delta^N (i, j),$$

$$\text{CR} = \frac{\sum_{i=1}^{N_g} \sum_{j=1}^{N_g} i \times j \times P_\delta^N (i, j) - \mu_X \mu_Y}{\sigma_X \sigma_Y}, \quad (8)$$

$$\text{ET} = -\sum_{i=1}^{N_g} \sum_{j=1}^{N_g} P_\delta^N (i, j) \log \left(P_\delta^N (i, j) \right),$$

where N_g denotes the number of gray level values; μ_X, μ_Y, σ_X, and σ_Y are the means and standard deviations of the marginal distribution with respect to $P_\delta^N (i, j)$.

2.4. Logistic Regression Model (LRM).
The LRM is a capable method for solving binary classification problems [29, 56–58]. The task at hand is to construct a decision

boundary that categorizes the input data into two distinctive regions. Therefore, given a vector of input data $x_i = x_{i1}, x_{i2}, ..., x_{iD}$, where D is the number of the features used for classification, the model is able to derive the class output y with either $y = 0$ (for the negative class of nonspall) and $y = 1$ (for the positive class of spall).

The probability of the positive class $h_\theta(x_i)$ derived by a LRM is given by [59]

$$h_\theta(x_i) = h_\theta(x_{i1}, x_{i2}, ..., x_{iD}) = \frac{1}{1 + \exp(-\eta_i)}$$

$$= \frac{1}{1 + \exp(-\theta^T x_i)}, \tag{9}$$

where $\eta_i = \theta_0 + \theta_1 x_{i1} + \theta_2 x_{i2} + \cdots + \theta_D x_{iD} = \theta^T x_i$. The vector $\theta = \theta_0, \theta_1, \theta_2, ..., \theta_D$ is the model parameter.

As a supervised learning approach, a set of training examples needs to be prepared so that the vector θ can be adapted during the model training phase. A LRM can be trained by either minimizing the least square loss function or maximizing the log likelihood function.

The least square loss function is given by [60]

$$\text{Loss}(\theta) = \sum_{i=1}^{M} (y_i - h_\theta(x_i))^2, \tag{10}$$

where M is the number of training data.

The log likelihood function is described as follows [61, 62]:

$$\text{Loss}(\theta) = \log(L(\theta)) = \sum_{i=1}^{M} y_i \log(h_\theta(x_i))$$

$$+ (1 - y_i)(1 - \log(h_\theta(x_i))). \tag{11}$$

A LRM can be trained via the stochastic gradient descent framework [29]. If the least square loss function is used, the update rule for adapting the model parameters is given by [60]

$$\theta_k = \theta_k + Lr \times (y - h_\theta(x_i)) \times h_\theta(x_i)(1 - h_\theta(x_i)) \times x_{i,k}, \tag{12}$$

where Lr denotes the learning rate parameter.

Meanwhile, if the log likelihood function is selected, the update rule used that compute θ is given by [61, 62]

$$\theta_k = \theta_k + Lr \times (y_i - h_\theta(x_i))x_{i,k}. \tag{13}$$

2.5. Adaptive Moment Estimation (Adam) Optimizer. Adam, proposed by [63], is designed as an algorithm for first-order gradient-based optimization of stochastic objective functions. This algorithm is relied on adaptive estimates of lower-order moments. Adam can be considered as an extension of the stochastic gradient descent employed to train machine learning models via an iterative weight updating process [64]. It is noted that the conventional stochastic gradient descent employs a constant learning rate (Lr) for all weight updates. Adam seeks for improving the

model training phase by adaptively fine-tuning the Lr parameter.

Adam harnesses information obtained from the average of the second moments of the gradients. In detail, this optimization algorithm computes an exponential moving average of the gradient and the square gradient. Moreover, a set of parameters (β_1 and β_2) is used to dictate the decay rates of these moving averages [64]. Via experimentation, it can be shown that the advantages of this optimizer include efficient computation, straight forward implementation, no memory requirements, and the capability of dealing with a large number of optimized parameters [63].

In order to implement Adam to optimize a LRM, it is necessary to compute the gradient (g_t). The gradient g_t in the case of using the least square loss function is given by [60]

$$g_t = \frac{\partial \text{Loss}(\theta)}{\partial \theta_k} = 2(y - h_\theta(x_i)) \times h_\theta(x_i) \times (1 - h_\theta(x_i)) \times x_{i,k}. \tag{14}$$

If the log likelihood function is employed, the gradient g_t is given by [22, 61, 62]

$$g_t = \frac{\partial \text{Loss}(\theta)}{\partial(\theta_k)} = (y_i - h_\theta(x_i))x_{i,k}. \tag{15}$$

Accordingly, the Adam procedure (illustrated in Algorithm 2) used for training a LRM can be performed iteratively with the following steps:

(i) Compute gradient g_t

(ii) Update the biased first and second raw moment estimates

(iii) Compute the bias-corrected moment estimates

(iv) Adapt the optimized parameters

3. Collection of Image Samples

The LRM used in this study belongs to the category of supervised machine learning methods. To train this LRM with the use of the aforementioned Adam optimizer, it is a requisite to prepare a set of training image samples as well as a set of testing image samples to verify the model construction phase. Therefore, this study has carried out field surveys at several high-rise buildings in Danang city (Vietnam) to collect a set of 600 image samples. Among them, 300 samples contain localized spall objects and 300 samples consist of nonlocalized spall objects. Notably, image samples of the two class of nonspall (class label = 0) and spall (class label = 1) have been assigned by a human inspector for the purposes of model training and testing. The Cannon EOS M10 (CMOS 18.0 MP) and Nikon D5100 (CMOS 16.2 MP) have been employed to collect image samples. In addition, the image size has been standardized to be 64×64 to facilitate the computation process. The image set has been collected so that a diverse background (e.g., cracks and stains) can be included. The collected image set is demonstrated in Figure 3.

```
Define step size a = 0.001
Define exponential decay rates β₁ = 0.9 and β₂ = 0.9999
Define the objective function f(θ)
Randomly initialize the searched variable θ
Assign m₀ = 0, v₀ = 0, and t = 0
(1)   While (θₜ not converged)
(2)         t = t + 1
(3)         Compute gradient: gₜ = ∇_θ fₜ(θₜ₋₁)
(4)         Update biased 1st moment estimate
(5)             mₜ = β₁mₜ₋₁ + (1 − β₁)gₜ
(6)         Update biased 2nd raw moment estimate
(7)             vₜ = β₂vₜ₋₁ + (1 − β₂)gₜ²
(8)         Calculate bias-corrected first moment estimate
(9)             m̂ₜ = mₜ/(1 − β₁ᵗ)
(10)        Calculate bias-corrected 2nd raw moment estimate
(11)            v̂ₜ = vₜ/(1 − β₂ᵗ)
(12)        Update the searched parameter
(13)            θₜ = θₜ₋₁ − α × (m̂ₜ/√v̂ₜ + ε)
(14)  End While
(15)  Return θₜ
```

ALGORITHM 2: The Adam optimization procedure.

(a)

(b)
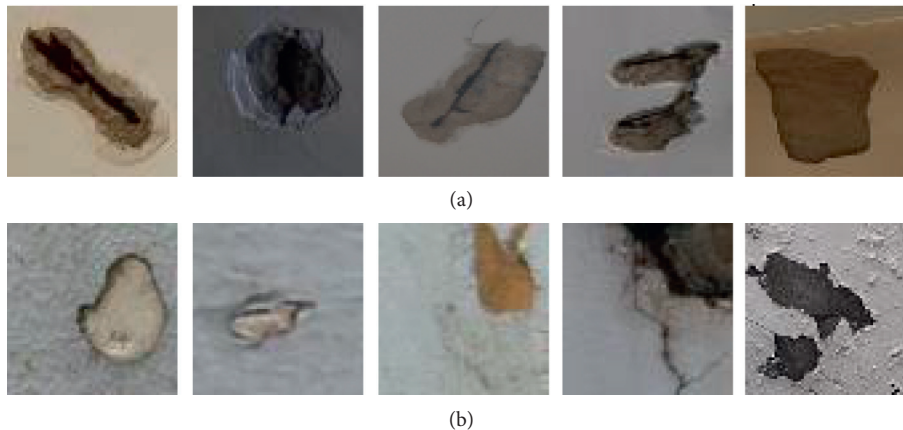
FIGURE 3: Demonstration of the collected image samples: (a) images containing localized spall objects and (b) images containing non-localized spall objects.

4. The Proposed Hybrid Approach of Image Processing and Machine Learning Approach for Automatic Detection of Concrete Spall

This section of the study aims at describing the structure of the proposed hybrid approach of image processing and machine learning used for recognizing localized spall object. The overall structure of the proposed approach is presented in Figure 4. It is noted that the hybrid model used for automatic concrete spall detection has been developed in Visual C#.NET environment (Framework 4.6.2) and implemented with the ASUS FX705GE–EW165T (Core i7 8750H, 8 GB Ram, 256 GB solid-state drive).

The model can be divided into several operational steps:

(i) Automatic ROI identification

(ii) Image texture computation

(iii) Machine learning-based pattern classification

4.1. Automatic Region of Interest (ROI) Identification. To deal with the diverse shapes of localized spall objects, this study relies on the techniques of GF to automatically identify ROIs that contains the potential defects of interest. It is noted that an image sample has been denoised by a median filter with a window size of 4 pixels and converted to a grayscale one. After the GFs with different orientations and radial frequency are computed, the principal component analysis (PCA) is performed to transform the set of GFs and reduce the data dimensionality (Figure 5). The number of the PCA transformed data is selected corresponding to 99% of cumulative variance explained. It is noted that the GF and the
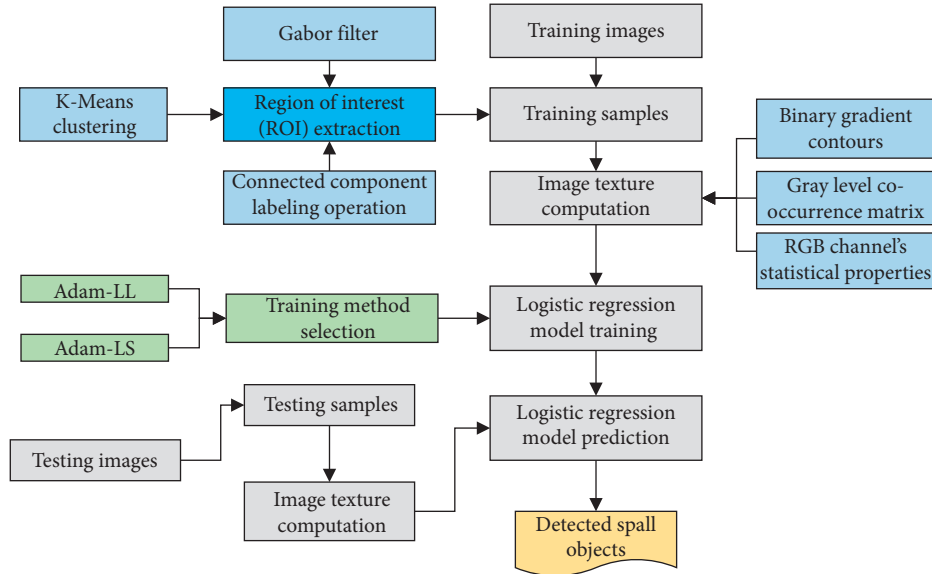
FIGURE 4: The proposed model structure.

PCA operations have been implemented via built-in functions provided by the Accord.NET Framework [65].

Based on the PCA result, the *k*-means clustering algorithm is used to segment the image sample. Via experimentation, the suitable number of clusters for the collected dataset is found to be 3. Subsequently, the morphological operation of filling and removing small objects are utilized to process the segmented image. Moreover, the operation of background removal is performed to remove redundant objects. In this study, an object within an image sample is considered to be background if its width or height is equal to that of the image sample.

Accordingly, each image cluster or segment is presented as a binary image. The connected component labeling algorithm [66] is then used to analyze the position of the binary-1 pixels and separate them into distinctive component regions. Essentially, all pixels having value binary 1 and are connected to each other are grouped into one object [38]. To remove crack objects, for each grouped pixels obtained from the connected component labeling analysis, an object slenderness index (OSI) is computed as follows:

$$\text{OSI} = \text{Max}\left(\frac{L_{OX}}{\mu_{OX}}, \frac{L_{OY}}{\mu_{OY}}\right), \qquad (16)$$

where L_{OX} and L_{OY} are the object lengths along the X axis and Y axis, respectively. μ_{OX} and μ_{OY} denote the mean thicknesses of the object along the X axis and Y axis, respectively.

If the OSI of an object is greater than a certain threshold (T_{OSI}), this object is classified as a crack. Via several trial-and-error experiments with the collected image sample, a suitable value for the threshold T_{OSI} is found to be 5. After the ROIs have been identified, operations of image convolution and cropping are employed to isolate the areas of interest within the image sample. The processes of ROI identification and isolation are demonstrated in Figures 6 and 7.
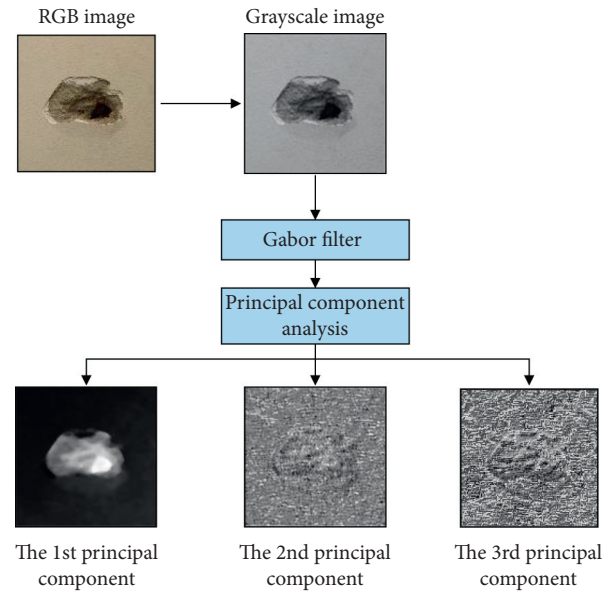
FIGURE 5: Image processed by the Gabor filter.

4.2. *Image Texture Computation.* Based on ROIs obtained from the previous step, image texture analysis consisting of statistical measurements of BGC, RGB channels, and GLCM is carried out. The BGC texture descriptor includes all of the three variants of single, double, and triple loops. Each of the variants yields a histogram which describes the texture property of an image sample. Accordingly, statistical indices including mean, standard deviation, skewness, kurtosis, and entropy can be computed for each histogram. Therefore, the BGC results in 15 numerical features. As mentioned earlier, the mean (μ_c), standard deviation (σ_c), skewness (S_c), kurtosis (K_c), entropy (E_c), and range (R_c) of the three color channels (R, G, and B) are used to represent the color features of image samples. Thus, there are $6 \times 3 = 18$ additional numerical features. Moreover, properties of the
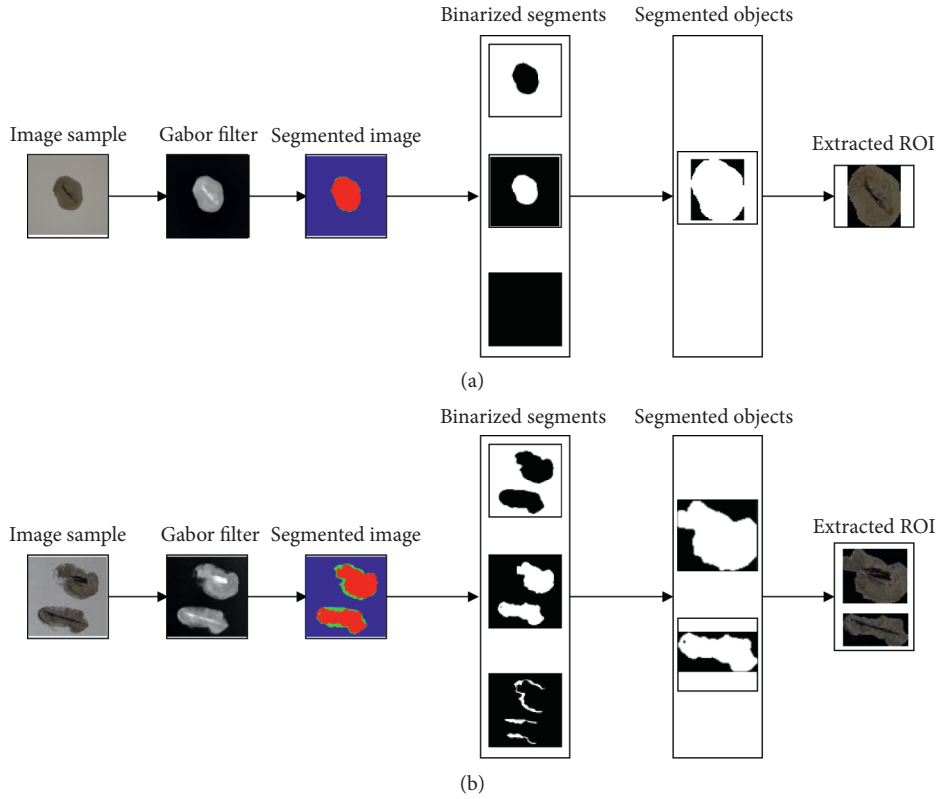
Figure 6: ROI extraction for images containing localized spall objects: (a) one object and (b) multiple objects.

GLCM including the four indices of angular second moment (AM), contrast (CO), correlation (CR), and entropy (ET) are used. It is noted that for each image sample, four GLCMs are established. Thus, the GLCM texture descriptor yields $4 \times 4 = 16$ features. In total, there are $15 + 18 + 16 = 49$ numerical features that can be extracted from the image texture computation process.

4.3. Pattern Classification Using LRM Trained by the Adam Optimizer. Using the extracted ROIs and the aforementioned texture descriptors, a dataset with 790 samples and 49 features can be constructed. This dataset contains 465 nonlocalized spall samples and 325 localized spall samples. As stated earlier, the output class is either 0 for the negative class and 1 for the positive class. Moreover, in order to standardize the input features' range, the numerical texture descriptors have been normalized by the Z-score equation as follows:

$$X_{ZN} = \frac{X_o - m_X}{s_X}, \tag{17}$$

where X_o and X_{ZN} represent the original and normalized input data, respectively. m_X and s_X are the mean and the standard deviation of the original input data, respectively.

Based on the aforementioned dataset, the LRM is trained with the Adam optimizer using the least square and log likelihood loss functions. These two LRM is denoted as Adam-LS and Adam-LL. It is noted that 90% of the collected dataset has been employed to construct the LRM model. Meanwhile, the rest of the dataset is reserved to verify the generalization capability of the model.

5. Research Findings and Discussion

As mentioned earlier, the whole collected dataset is divided into two subsets: a training set (90%) and a testing set (10%). Moreover, to diminish the effect of randomness brought about by data sampling and to assess the generalization capability of the integrated method reliably, the data sampling process has been repeated 20 times. A partitioned datasets used for model training and testing are demonstrated in Table 1. In addition, the LRM trained with the stochastic gradient descent algorithm with the least square and log likelihood loss function are employed as benchmark models. The stochastic gradient descent models coupled with the former and later loss function are denoted as LS-LR and LL-LR, respectively. Furthermore, the two LRMs trained with the Adam optimizer are denoted as Adam-LS-LR and Adam-LL-LR. All of the LRMs have been trained with 1000 iterations.

In addition, the classification accuracy rate (CAR), precision, recall, negative predictive value (NPV), and F1 score are computed to quantify the model predictive accuracy. These performance measurement indices are provided as follows [67]:
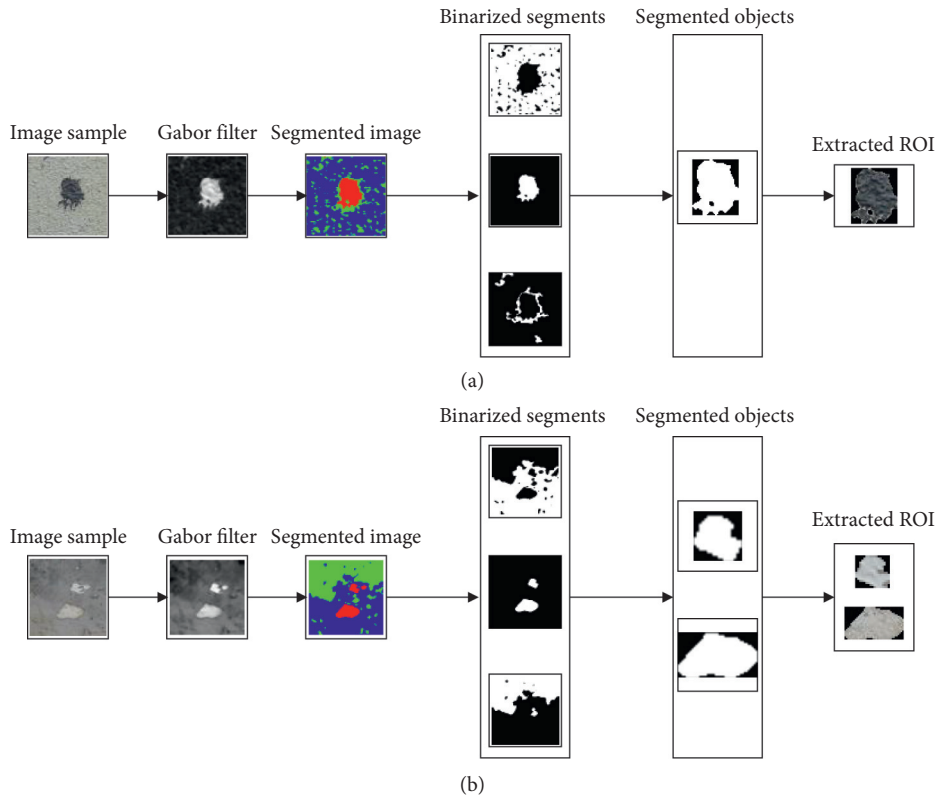
FIGURE 7: ROI extraction for images containing nonlocalized spall objects: (a) one object and (b) multiple objects.

TABLE 1: Demonstration of the collected dataset.

Set	Case	IF1	IF2	IF3	IF4	. . .	IF46	IF47	IF48	IF49	Class label
	1	39.64	37.27	34.53	34.33	. . .	0.10	745.72	0.00	6.47	0
	2	12.92	11.63	9.60	18.34	. . .	0.17	497.69	0.00	3.82	0
	3	68.21	65.76	59.38	40.67	. . .	0.03	770.71	0.00	6.82	0
	4	44.78	45.60	43.86	56.59	. . .	0.29	1835.96	0.00	4.39	0
	5	65.40	64.84	66.24	60.02	. . .	0.12	2458.40	0.00	4.12	0
Training	. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .
	707	68.85	58.73	45.76	34.57	. . .	0.01	824.92	0.00	6.77	1
	708	55.85	51.50	46.05	26.86	. . .	0.01	762.79	0.00	7.48	1
	709	51.27	47.45	42.09	35.80	. . .	0.05	928.80	0.00	6.93	1
	710	60.32	55.98	49.76	31.67	. . .	0.01	1340.14	0.00	6.67	1
	711	73.54	65.91	62.20	44.11	. . .	0.02	1180.24	0.00	7.82	1
	1	56.26	58.54	61.95	25.44	. . .	0.01	498.38	0.00	7.27	0
	2	93.11	95.22	103.83	41.81	. . .	0.10	1353.60	0.00	3.32	0
	3	85.93	83.79	93.25	49.92	. . .	0.07	3975.50	0.00	3.95	0
	4	58.64	59.44	64.39	55.01	. . .	0.18	968.76	0.00	5.57	0
	5	83.84	84.93	88.35	68.18	. . .	0.14	493.65	0.00	5.54	0
Testing	. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .
	75	62.98	47.68	28.07	32.93	. . .	0.03	244.76	0.00	7.79	1
	76	46.17	44.28	39.69	33.57	. . .	0.08	467.10	0.00	6.35	1
	77	51.47	46.08	37.80	20.61	. . .	0.01	370.11	0.00	8.62	1
	78	34.17	30.02	24.89	29.42	. . .	0.10	413.32	0.00	6.74	1
	79	44.89	37.54	27.15	56.16	. . .	0.32	528.69	0.00	5.10	1

TABLE 2: Prediction result comparison.

Phase	Indices	Adam-LL-LR		LL-LR		Adam-LS-LR		LS-LR	
		Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.
Training	CAR (%)	85.27	1.76	84.52	3.82	72.52	1.69	71.81	2.52
	TP	246.95	12.48	239.70	26.87	271.85	8.71	265.65	14.59
	TN	359.30	13.68	361.25	14.25	243.75	13.76	244.90	11.57
	FP	46.85	12.77	53.15	26.97	22.50	7.68	28.25	15.63
	FN	57.90	14.13	56.90	14.71	172.90	12.92	172.20	10.72
	Precision	0.84	0.04	0.82	0.09	0.92	0.03	0.90	0.05
	Recall	0.81	0.04	0.81	0.04	0.61	0.02	0.61	0.02
	NPV	0.86	0.03	0.86	0.03	0.58	0.03	0.59	0.03
	F1 score	0.82	0.02	0.81	0.06	0.74	0.01	0.73	0.03
Testing	CAR (%)	85.32	4.64	81.90	4.06	72.03	5.37	70.82	5.92
	TP	26.90	5.19	25.25	3.94	27.60	3.29	28.10	6.03
	TN	40.50	5.68	39.45	4.78	29.30	3.45	27.85	5.17
	FP	4.30	2.26	6.90	2.62	3.05	2.42	3.00	2.02
	FN	7.30	2.76	7.40	2.91	19.05	4.34	20.05	3.73
	Precision	0.86	0.07	0.79	0.08	0.90	0.07	0.90	0.09
	Recall	0.79	0.08	0.78	0.06	0.59	0.07	0.58	0.09
	NPV	0.85	0.06	0.84	0.07	0.61	0.08	0.58	0.07
	F1 score	0.82	0.06	0.78	0.05	0.71	0.06	0.70	0.08

$$CAR = \frac{TP + TN}{TP + TN + FP + FN} \times 100\%,$$

$$Precision = \frac{TP}{TP + FP},$$

$$Recall = \frac{TP}{TP + FN},$$ (18)

$$NPV = \frac{TN}{TN + FN},$$

$$F1\ score = \frac{2TP}{2TP + FP + FN},$$

where FN, FP, TP, and TN denote the number of false-negative, false-positive, true-positive, and true-negative samples, respectively.

The experimental results obtained from the repetitive data sampling with 20 runs are reported in Table 2. As can be seen from this table, the Adam-LL-LR has achieved the best predictive accuracy in both of the training (CAR = 85.25%, precision = 0.84, recall = 0.81, NPV = 0.86, and F1 score = 0.82) and testing phases (CAR = 85.32%, precision = 0.86, recall = 0.79, NPV = 0.85, and F1 score = 0.82). Since the prediction performances obtained from the training and testing phases of the Adam-LL-LR are relatively similar, it can be shown that this model has not suffered from overfitting. In addition, the LL-LR model is the second best approach (with CAR = 81.90% and F1 score = 0.78), followed by the Adam-LS-LL (with CAR = 72.03% and F1 score = 0.71) and the LS-LR (with CAR = 70.82% and F1 score = 0.70). Herein, the index of the F1 score is emphasized because it presents the harmonic mean of the precision and recall.

The training and testing performances of the employed models are graphically presented in Figures 8 and 9. The boxplot shown in Figure 10 demonstrates the testing
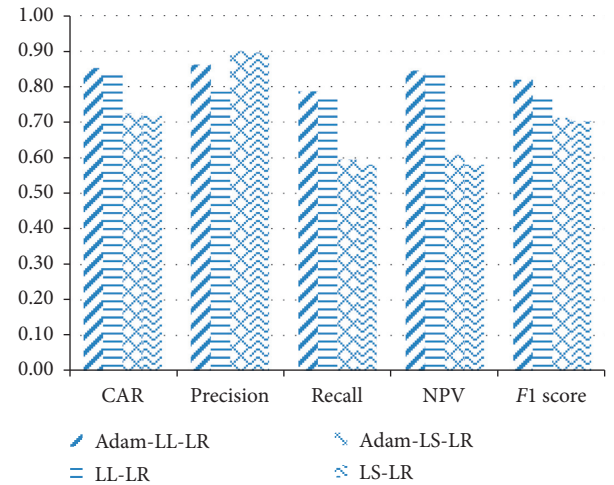
FIGURE 8: Performance measurement indices for the training phase.

performances of LRMs. In addition, to confirm the statistical difference of each pair of the localized spall detection models, the Wilcoxon signed-rank test with a significance level (p value) = 0.05 is employed in this section of the study. The test outcomes are reported in Table 3. Observably, experimental results show that all of the p values are lower than the significance level. Thus, the null hypothesis shows that the performances of the two models under testing are statistically indifferent and can be confidently rejected. This hypothesis test asserts the superiority of the Adam-LL-LR model over other benchmark approaches.

Based on the experimental result, the Adam-LL-LR model is best suited for the collected dataset at hand. The performance of this model is further studied in this section. Illustrations of correctly recognized spall objects yielded by Adam-LL-LR are presented in Figure 11. As can be observed, the model can deliver accurate detection results in the
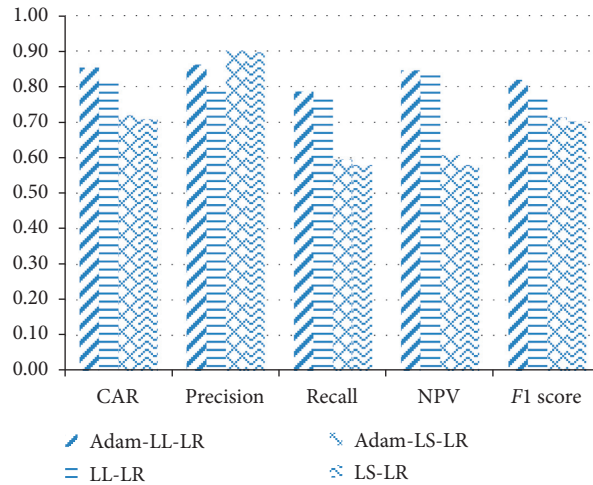
FIGURE 9: Performance measurement indices for the testing phase.

(a)

(b)
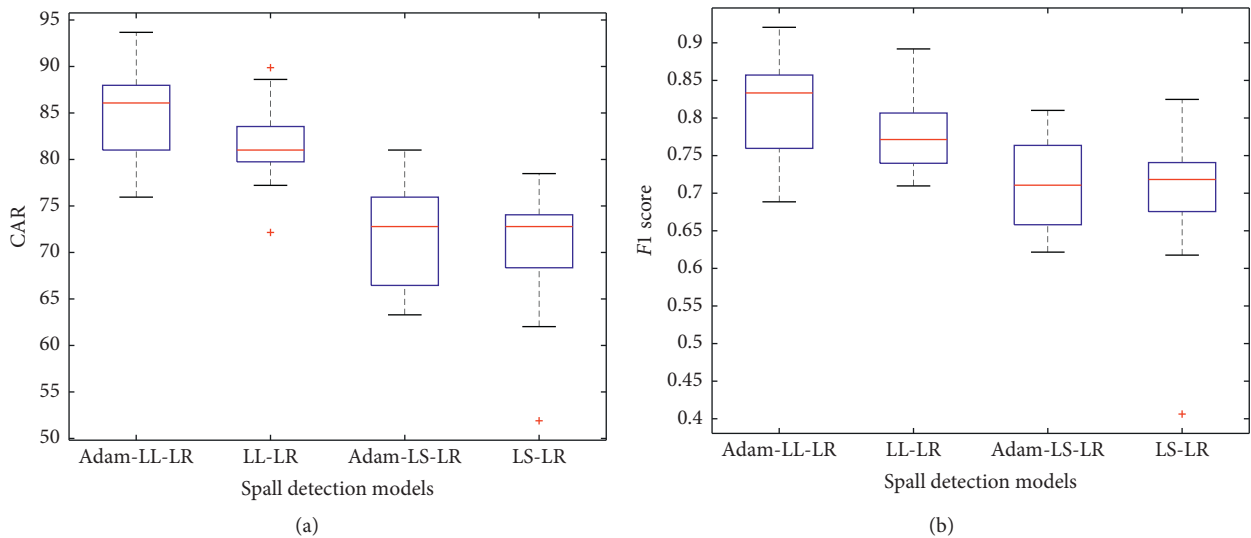
FIGURE 10: Model performances obtained from the repetitive data subsampling process: (a) CAR (%) and (b) $F1$ score.

TABLE 3: Wilcoxon signed-rank test results.

Indices	Model comparison	Test outcome	p value
	Adam-LL-LR vs. LL-LR	Significant	0.0184
CAR (%)	Adam-LL-LR vs. Adam-LS-LR	Significant	0.0001
	Adam-LL-LR vs. LS-LR	Significant	0.0001
	Adam-LL-LR vs. LL-LR	Significant	0.0400
$F1$ score	Adam-LL-LR vs. Adam-LS-LR	Significant	0.0004
	Adam-LL-LR vs. LS-LR	Significant	0.0001

presence of a window (Figure 11(a)) and a minor defect on the mortar surface (Figure 11(b)). Notably, the localized spall objects can still be located well in the cases that there are crack objects in the captured scenes (Figures 11(c)–11(e)). Furthermore, Adam-LL-LR has also performed well in the cases that there are multiple spall objects in the image samples (Figures 11(f) and 11(g)). In addition, the proposed

Adam-LL-LR model can be used to quantify the percentage of damaged areas found in image samples; the computation results are demonstrated in Figure 12.

Nevertheless, as shown in Figure 13, the newly developed model has made incorrect detection results in the cases of complex background. As observed in Figure 13(a), an area in the background has the texture property similar to that of the

Original image	Segmented image	Binary-detected object	Detected object with positive probability (Prob.)

(a)

(b)

(c)

(d)

(e)

(f)

(g)
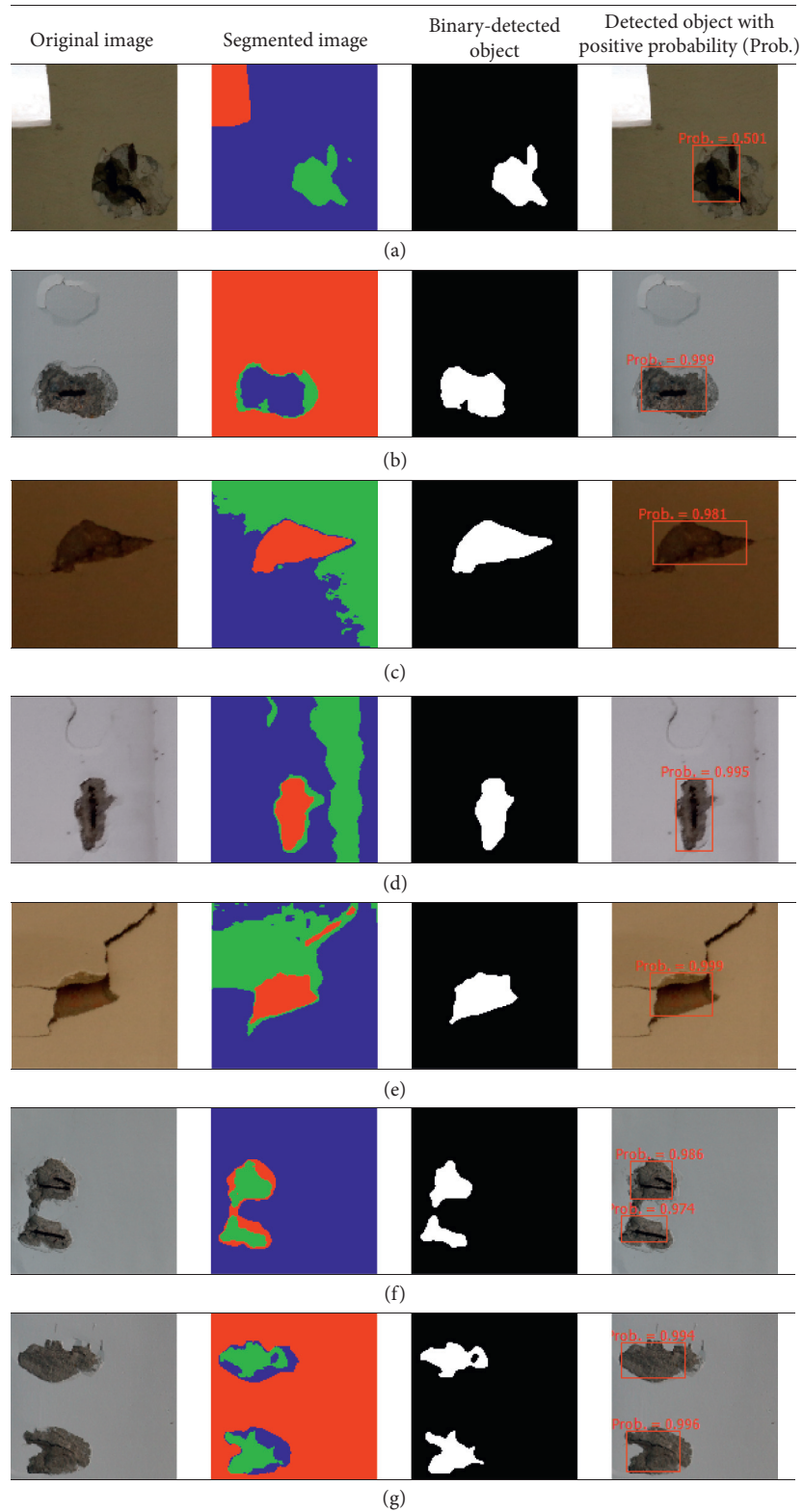
FIGURE 11: Illustrations of correctly classified cases.

Original image	Segmented image	Detected objects	Computation results

Percentage of damaged area: 8.70%

Percentage of damaged area of the 1st object: 4.88%

Percentage of damaged area of the 2nd object: 4.84%

FIGURE 12: Demonstration of the calculation of the percentage of damaged areas.

Original image	Segmented image	Binary-detected object	Falsely detected/undetected object

(a)

Undetected object

(b)

Undetected object

(c)

FIGURE 13: Illustrations of incorrectly classified cases.

spall object. This can lead to a false-positive detection. Complex background (Figure 13(b)) and irregular lighting conditions (Figure 13(c)) also tend to reduce the model accuracy. These phenomena can lead to false-negative cases.

6. Concluding Remarks

Localized spall is a common defect observed on surfaces of reinforced concrete elements. Accurate detection of this damage is crucial during the phase of the periodic structural heath survey. This study has developed and verified a computer vision-based approach for automating the task of localized spall recognition. The newly developed model is a hybridization of image processing and machine learning approaches. Image processing methods of the GF coupled with k-means clustering and morphological analyses are used to automatically identify the ROIs that potentially contain the defect. The BGC, GLCM, and color channels' properties are employed as texture descriptors. Based on the computed image texture, the LRM optimized by the state-of-the-art Adam is used to construct a decision boundary that separate the data samples into two regions of nonlocalized spall and localized spall. Experimental results show that the LRM trained by the Adam optimizer can deliver the most

desired prediction accuracy. Therefore, the proposed integrated model can be a useful tool to assist building maintenance agencies in the task of evaluating structure heath condition.

Data Availability

The image dataset used to support the findings of this study has been deposited in the repository of GitHub: https://github.com/NhatDucHoang/LocalizedSpallDetection_AdamLRM.

Conflicts of Interest

The author declares that there are no conflicts of interest.

Acknowledgments

References

[1] M. R. Jahanshahi and S. F. Masri, "Adaptive vision-based crack detection using 3D scene reconstruction for condition assessment of structures," *Automation in Construction*, vol. 22, pp. 567–576, 2012.

[2] B. A. Soeleman, M. V. A. D. Silva, M. Akkari, and C. Santili, "Accidents due to falls from roof slabs," *Sao Paulo Medical Journal*, vol. 131, no. 3, pp. 153–157, 2013.

[3] S. M. Verhulst, *Spalling Is Bad, Falling Is Worse*, Pieresearch, Arlington, TX, USA, 2020, https://pieresearchcom/spalling-is-bad-falling-is-worse/.

[4] M. Duckett and G. Pestine, *Concrete Spalling Robson Forensic*, https://wwwrobsonforensiccom/articles/concrete-spalling-expert-article/, 2020.

[5] C. Koch, K. Georgieva, V. Kasireddy, B. Akinci, and P. Fieguth, "A review on computer vision based defect detection and condition assessment of concrete and asphalt civil infrastructure," *Advanced Engineering Informatics*, vol. 29, no. 2, pp. 196–210, 2015.

[6] R. Zaurin, "Structural health monitoring with emphasis on computer vision, damage indices, and statistical analysis," https://starslibraryucfedu/cgi/viewcontentcgi?article=4853&context=etdElectronic theses and Dissertations, University of Central Florida, Orlando, FL, USA, 2009, https://starslibraryucfedu/cgi/viewcontentcgi?article=4853&context=etdElectronic theses and Dissertations.

[7] I. Abdel-Qader, S. Pashaie-Rad, O. Abudayyeh, and S. Yehia, "PCA-Based algorithm for unsupervised bridge crack detection," *Advances in Engineering Software*, vol. 37, no. 12, pp. 771–778, 2006.

[8] M. O'Byrne, F. Schoefs, B. Ghosh, and V. Pakrashi, "Texture analysis based damage detection of ageing," *Infrastructural Elements Computer-Aided Civil and Infrastructure Engineering*, vol. 28, pp. 162–177, 2013.

[9] D. Lattanzi and G. R. Miller, "Robust automated concrete damage detection algorithms for field applications," *Journal of Computing in Civil Engineering*, vol. 28, no. 2, pp. 253–262, 2014.

[10] D. B. Agyemang and M. Bader, "Surface crack detection using hierarchal convolutional neural network," in *Advances in Computational Intelligence Systems*, pp. 173–186, Springer International Publishing, Cham, Switzerland, 2020.

[11] Z. Chen and T. C. Hutchinson, "Image-based framework for concrete surface crack monitoring and quantification," *Advances in Civil Engineering*, vol. 2010, Article ID 215295, 18 pages, 2010.

[12] G. K. Choudhary and S. Dey, "Crack detection in concrete surfaces using image processing, fuzzy logic, and neural networks," in *Proceedings of the 2012 IEEE Fifth International Conference on Advanced Computational Intelligence (ICACI)*, pp. 404–411, Nanjing, China, October 2012.

[13] S. Dorafshan, R. J. Thomas, and M. Maguire, "Benchmarking image processing algorithms for unmanned aerial system-assisted crack detection in concrete structures," *Infrastructures*, vol. 4, no. 2, p. 19, 2019.

[14] N.-D. Hoang, "Detection of surface crack in building structures using image processing technique with an improved otsu method for image thresholding," *Advances in Civil Engineering*, vol. 2018, Article ID 3924120, 10 pages, 2018.

[15] N.-D. Hoang and Q.-L. Nguyen, "Metaheuristic optimized edge detection for recognition of concrete wall cracks: a comparative study on the performances of roberts, prewitt, Canny, and sobel algorithms," *Advances in Civil Engineering*, vol. 2018, Article ID 7163580, 16 pages, 2018.

[16] H. Kim, J. Lee, E. Ahn, S. Cho, M. Shin, and S.-H. Sim, "Concrete crack identification using a UAV incorporating hybrid image processing," *Sensors*, vol. 17, no. 9, p. 2052, 2017.

[17] S. Li and X. Zhao, "Image-based concrete crack detection using convolutional neural network and exhaustive search technique," *Advances in Civil Engineering*, vol. 2019, Article ID 6520620, 12 pages, 2019.

[18] M.-M. Naddaf-Sh, S. Hosseini, J. Zhang, N. A. Brake, and H. Zargarzadeh, "Real-time road crack mapping using an optimized convolutional neural network," *Complexity*, vol. 2019, Article ID 2470735, 17 pages, 2019.

[19] W. Song, G. Jia, H. Zhu, D. Jia, and L. Gao, "Automated pavement crack damage detection using deep multiscale convolutional features," *Journal of Advanced Transportation*, vol. 2020, Article ID 6412562, 11 pages, 2020.

[20] B. Wang, Y. Li, W. Zhao, Z. Zhang, Y. Zhang, and Z. Wang, "Effective crack damage detection using multilayer sparse feature representation and incremental extreme learning machine," *Applied Sciences*, vol. 9, no. 3, p. 614, 2019.

[21] T. Dawood, Z. Zhu, and T. Zayed, "Machine vision-based model for spalling detection and quantification in subway networks," *Automation in Construction*, vol. 81, pp. 149–160, 2017.

[22] N.-D. Hoang, Q.-L. Nguyen, and X.-L. Tran, "Automatic detection of concrete spalling using piecewise linear stochastic gradient descent logistic regression and image texture analysis," *Complexity*, vol. 2019, Article ID 5910625, 14 pages, 2019.

[23] C. Koch, S. G. Paal, A. Rashidi, Z. Zhu, M. König, and I. Brilakis, "Achievements and challenges in machine vision-based inspection of large concrete structures," *Advances in Structural Engineering*, vol. 17, no. 3, pp. 303–318, 2014.

[24] H. Wu, X. Ao, Z. Chen, C. Liu, Z. Xu, and P. Yu, "Concrete spalling detection for metro tunnel from point cloud based on roughness," *Descriptor Journal of Sensors*, vol. 2019, Article ID 8574750, 12 pages, 2019.

[25] S. German, I. Brilakis, and R. DesRoches, "Rapid entropy-based detection and properties measurement of concrete spalling with machine vision for post-earthquake safety assessments," *Advanced Engineering Informatics*, vol. 26, no. 4, pp. 846–858, 2012.

[26] P. Prasanna, K. J. Dana, N. Gucunski et al., "Automated crack detection on concrete bridges," *IEEE Transactions on Automation Science and Engineering*, vol. 13, no. 2, pp. 591–599, 2016.

[27] M.-K. Kim, H. Sohn, and C.-C. Chang, "Localization and quantification of concrete spalling defects using terrestrial laser scanning," *Journal of Computing in Civil Engineering*, vol. 29, Article ID 04014086, 2015.

[28] N.-D. Hoang, "Image processing-based recognition of wall defects using machine learning approaches and steerable filters," *Computational Intelligence and Neuroscience*, vol. 2018, Article ID 7913952, 18 pages, 2018.

[29] N.-D. Hoang, "Automatic detection of asphalt pavement raveling using image texture based feature extraction and stochastic gradient descent logistic regression," *Automation in Construction*, vol. 105, Article ID 102843, 2019.

[30] G. Yao, F. Wei, Y. Yang, and Y. Sun, "Deep-learning-based bughole detection for concrete surface image," *Advances in Civil Engineering*, vol. 2019, Article ID 8582963, 12 pages, 2019.

[31] Y. Li, X. Li, H. Wang, S. Wang, S. Gu, and H. Zhang, "Exposed aggregate detection of stilling basin slabs using attention U-net network," *KSCE Journal of Civil Engineering*, vol. 24, no. 6, p. 1740, 2020.

[32] J. K. Dey, Z. Su, J. Wu, P. S. Tan, X. Mao, and Y. H. Wang, "Anomaly detection of defects on concrete structures with the convolutional autoencoder," *Advanced Engineering Informatics*, vol. 45, Article ID 101105, 2020.

[33] L. Wang, K. i. Kawaguchi, and P. Wang, "Damaged ceiling detection and localization in large-span structures using convolutional neural networks," *Automation in Construction*, vol. 116, Article ID 103230, 2020.

[34] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning (Adaptive Computation and Machine Learning Series)*, The MIT Press, Cambridge, MA, USA, 2016, ISBN-10: 026203561.

[35] P. Kim, *MatLab Deep Learning with Machine Learning, Neural Networks and Artificial Intelligence*, Apress, New York, NY, USA, 2017.

[36] S. Skansi, *Introduction to Deep Learning—from Logical Calculus to Artificial Intelligence*, Springer International Publishing, New York, NY, USA, 2018.

[37] A. S. Moghaddam, E. R. Azar, Y. Mejias, and H. Bell, "Estimating stripping of asphalt coating using k-means clustering and machine learning–based classification," *Journal of Computing in Civil Engineering*, vol. 34, Article ID 04019044, 2020.

[38] L. G. Shapiro and G. C. Stockman, *Computer Vision*, Prentice-Hall, Upper Saddle River, NJ, USA, 2001.

[39] M. Sonka, V. Hlavac, and R. Boyle, *Image Processing, Analysis, and Machine Vision*, Cengage Learning, Printed in the United States of America, Boston, MA, USA, 2013, ISBN-10: 1-133-59360-7.

[40] S. Molaei and M. E. Shiri Ahmad Abadi, "Maintaining filter structure: a Gabor-based convolutional neural network for image analysis," *Applied Soft Computing*, vol. 88, Article ID 105960, 2020.

[41] A. Premana, A. P. Wijaya, and M. A. Soeleman, "Image segmentation using Gabor filter and K-means clustering method," in *Proceedings of the 2017 International Seminar on Application for Technology of Information and Communication (iSemantic)*, pp. 95–99, Semarang, Indonesia, October 2017.

[42] Z. Eduardo, G. G. B. Jaime, M. Roberto, and L. José, "Road crack detection using visual features extracted by gabor filters," *Computer-Aided Civil and Infrastructure Engineering*, vol. 29, pp. 342–358, 2014.

[43] K. Hammouda, *Texture Segmentation Using Gabor Filters Technical Report*, University of Waterloo, Waterloo, Canada, 2000, http://wwwpamiuwaterlooca/pub/hammouda/sd775-paperpdf.

[44] N. C. Kim and H. J. So, "Directional statistical Gabor features for texture classification," *Pattern Recognition Letters*, vol. 112, pp. 18–26, 2018.

[45] A. K. Jain and F. Farrokhnia, "Unsupervised texture segmentation using Gabor filters," *Pattern Recognition*, vol. 24, no. 12, pp. 1167–1186, 1991.

[46] J. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proceedings of the Fifth Berkeley Sympos Math Statist and Probability (Berkeley, Calif, 1965/66) Vol I: Statistics*, pp. 281–297, Univ California Press, Berkeley, CA, USA, 1967.

[47] R. Duda, P. Hart, and D. Stock, *Pattern Classification*, John Wiley & Sons, Hoboken, NJ, USA, 2001, ISBN-13: 978-0471056690.

[48] A. Fernández, M. X. Álvarez, and F. Bianconi, "Image classification with binary gradient contours," *Optics and Lasers in Engineering*, vol. 49, no. 9-10, pp. 1177–1184, 2011.

[49] F. Alaei, A. Alaei, U. Pal, and M. Blumenstein, "A comparative study of different texture features for document image retrieval," *Expert Systems with Applications*, vol. 121, pp. 97–114, 2019.

[50] S. Theodoridis and K. Koutroumbas, *Pattern Recognition*, Academic Press, Printed in the United States of America, Cambridge, MA, USA, 2009, ISBN 978-1-59749-272-0.

[51] R. M. Haralick, K. Shanmugam, and I. H. Dinstein, "Textural features for image classification," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-3, no. 6, pp. 610–621, 1973.

[52] R. M. Haralick and L. G. Shapiro, *Computer and Robot Vision*, Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1992, ISBN 0201569434.

[53] G. M. Hadjidemetriou, P. A. Vela, and S. E. Christodoulou, "Automated pavement patch detection and quantification using support vector machines," *Journal of Computing in Civil Engineering*, vol. 32, Article ID 04017073, 2018.

[54] N.-D. Hoang, "Image processing-based pitting corrosion detection using metaheuristic optimized multilevel image thresholding and machine-learning approaches," *Mathematical Problems in Engineering*, vol. 2020, Article ID 6765274, 19 pages, 2020.

[55] F. Tomita and S. Tsuji, *Computer Analysis of Visual Textures*, Springer Science + Business Media, New York, NY, USA, 1990, ISBN 978-1-4612-8832-9.

[56] M. Andrejiova, A. Grincova, and D. Marasova, "Failure analysis of rubber composites under dynamic impact loading by logistic regression," *Engineering Failure Analysis*, vol. 84, pp. 311–319, 2018.

[57] H. C. Chan, C. C. Chang, P. A. Chen, and J. T. Lee, "Using multinomial logistic regression for prediction of soil depth in an area of complex topography in Taiwan," *Catena*, vol. 176, pp. 419–429, 2019.

[58] W. W. Piegorsch, *Statistical Data Analytics: Foundations for Data Mining, Informatics, and Knowledge Discovery*, John Wiley & Sons, Hoboken, NJ, USA, 2015, ISBN 978-1-118-61965-0.

[59] A. Agresti, *An Introduction to Categorical Data Analysis*, John Wiley & Sons, Hoboken, NJ, USA, 2019, ISBN 9781119405283.

[60] S. J. Russell and P. Norvig, *Artificial Intelligence—A Modern Approach*, Pearson Education, Upper Saddle River, NJ, USA, 2010.

[61] M. Gormley, *Logistic Regression*, https://www.cs.cmu.edu/%7Emgormley/courses/10701-f16/slides/lecture5.pdf, 2016.

[62] A. Ng, *Lecture Notes*, Stanford University, Kunnamangalam, Kerala, 2018, http://cs229.stanford.edu/notes/cs229-notes1.pdf.

[63] D. P. Kingma and J. Ba, "Adam: a method for stochastic optimization," in *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*, San Diego, CA, USA, May 2015.

[64] J. Brownlee, *Gentle Introduction to the Adam Optimization Algorithm for Deep Learning Machine Learning Mastery*, https://machinelearningmasterycom/adam-optimization-algorithm-for-deep-learning/, 2017.

[65] Accord, "Accord. NET Framework," 2019, http://accord-frameworknet/.

[66] R. M. Haralick, "Some neighborhood operators," in *Real-Time Parallel Computing: Imaging Analysis*, M. Onoe, K. Preston, and A. Rosenfeld, Eds., Springer, Boston, MA, USA, pp. 11–35, 1981.

[67] D. Tien Bui, N.-D. Hoang, H. Nguyen, and X.-L. Tran, "Spatial prediction of shallow landslide using Bat algorithm optimized machine learning approach: a case study in Lang Son Province, Vietnam," *Advanced Engineering Informatics*, vol. 42, Article ID 100978, 2019.

Research Article

Implementation of Parallel K-Means Algorithm to Estimate Adhesion Failure in Warm Mix Asphalt

Mohammad Nishat Akhtar [iD],[1] **Waseem Ahmed** [iD],[2] **Muhammad Rafiq Kakar** [iD],[3,4] **Elmi Abu Bakar** [iD],[1] **A. R. Othman** [iD],[5] **and Moises Bueno** [iD][3]

[1]*School of Aerospace Engineering, Universiti Sains Malaysia, Nibong Tebal 14300, George Town, Penang, Malaysia*
[2]*Faculty of Computing and IT, King Abdulaziz University, Jeddah, Saudi Arabia*
[3]*Empa Swiss Federal Laboratories for Materials Science and Technology, CH, Dübendorf 8600, Switzerland*
[4]*Department of Architecture, Wood and Civil Engineering, Bern University of Applied Sciences (BFH), CH-3012 Bern, Switzerland*
[5]*Department of Mechanical Engineering, Universiti Teknologi PETRONAS, Seri Iskandar 32610, Perak, Malaysia*

Correspondence should be addressed to Elmi Abu Bakar; meelmi@usm.my and A. R. Othman; rahim.othman@utp.edu.my

Warm Mix Asphalt (WMA) and Hot Mix Asphalt (HMA) are prepared at lower temperatures, making it more susceptible to moisture damage, which eventually leads to stripping due to the adhesion failure. Moreover, the assessment of the adhesion failure depends on the expertise of the investigator's subjective visual assessment skills. Nowadays, image processing has gained popularity to address the inaccuracy of visual assessment. To attain high accuracy from image processing algorithms, the loss of pixels plays an essential role. In high-quality image samples, processing takes more execution time due to the greater resolution of the image. Therefore, the execution time of the image processing algorithm is also an essential aspect of quality. This manuscript proposes a parallel k means for image processing (PKIP) algorithm using multiprocessing and distributed computing to assess the adhesion failure in WMA and HMA samples subjected to three different moisture sensitivity tests (dry, one, and three freeze-thaw cycles) and fractured by indirect tensile test. For the proposed experiment, the number of clusters was chosen as ten ($k = 10$) based on k value and cost of k means function was computed to analyse the adhesion failure. The results showed that the PKIP algorithm decreases the execution time up to 30% to 46% if compared with the sequential k means algorithm when implemented using multiprocessing and distributed computing. In terms of results concerning adhesion failure, the WMA specimens subjected to a higher degree of moisture effect showed relatively lower adhesion failure compared to the Hot Mix Asphalt (HMA) samples when subjected to different levels of moisture sensitivity.

1. Introduction

The image processing method has been widely used as a nondestructive system to evaluate 2D or 3D geometry in numerous scientific fields [1]. In the field of civil engineering, image processing has been successfully applied in multiple applications such as pavement distress assessment, site evaluation using satellite imaging, and analysis of crack propagation and microstructures in cement-based materials [2]. Concerning asphalt pavement binders, a number of computer-vision based system has been developed, which is broadly categorized into field assessment and laboratory applications. Concerning field assessment, algorithms were developed for the identification of pavement distress type and size [2–5]. Nonetheless, these algorithms also allow identifying the fractured surfaces of asphalt mixtures such as broken aggregates or the adhesion and cohesion failures at the interface of failed specimens [1–3].

In this work, the type and degree of failure due to the moisture effect is evaluated by image analysis for a series of Warm Mix Asphalt (WMA) mixtures. WMA technology uses lower temperatures in comparison with conventional Hot Mix Asphalt (HMA). Even though in contrast with conventional Hot Mix Asphalt (HMA), the WMA

technologies are more environmentally friendly and cost-effective; however, due to the lower production temperature, the WMA mixtures have shown less resistance against moisture damage [6–8]. Concerning distress or stripping identification using a computer vision-based system, it is essential to know that the distressed/stripped area pixel always possesses a lower pixel intensity compared to the unstripped area [9]. Conventional tests carried out to identify moisture damage region of interest (ROI) in asphalt mixtures include the use of visual assessment depending on the perception of the investigator [10], [11]. In such a scenario, the investigator might miss out to select some part of the ROI, which finally hampers the end result. Therefore, to further understand the mechanisms, it seems necessary to use an image segmentation algorithm that can quantify these failures more accurately. It is deemed necessary to note that these types of image processing involve high-performance computation because the high-end image processing cannot be handled efficiently on single computing node. In this regard, a framework is needed that allows the researchers to concentrate on the image processing tasks and refrains by getting them involved into complicated details of distributed computing. Additionally, the framework should provide the researchers with the familiar image processing tools.

This work proposes an implementation of clustering on high-resolution images of WMA broken samples using a parallel k means for image processing (PKIP) algorithm. To implement the k means algorithm parallelly, multiprocessing on a single node and Map-Reduce based programming on multiple nodes have been used [12].

An essential factor in the visual assessment of the stripping is concerning to its representation. In terms of pixel value intensity, the stripped and the unstripped area must have a significant difference. In order to classify the pixels with respect to different intensities or color similarities, the k means clustering algorithm is widely used. Using the k means algorithm, n observations can be segregated into k classes based on a particular mean value. The proximity of each observation to the cluster is iteratively processed using the nearest mean. The variant of k means comprises k median and k medoid (see Appendix section). It is worth to be noted that k median is best suited for local optimization problem, whereas k means algorithm is tailored for both, i.e., global and local optimization [13]. On the other hand, k medoid has high algorithmic time complexity $o(n^2 * k * i)$ (see Appendix section), and thus, it is compute intensive compared to k means $o(n^2 * k * i)$ [14, 15].

After this short introduction, the manuscript is arranged as follows, Section 2 describes the previous work carried out on asphalt mixture analysis using conventional image processing techniques and also describes the challenges in terms of processing time required to analyse the high-end image datasets using sequential k means clustering. Section 3 starts with WMA specimens' preparation and proposes the PKIP algorithm for central processing unit- (CPU-) based multiprocessing execution and Map-Reduce based distributed computing, followed by a thresholding process to assess the adhesion failure. Section 4 presents the results in terms of the accuracy and performance of the proposed PKIP

algorithm. Section 5 presents the discussion related to the outcome of the results. Finally, Section 6 summarizes the most significant conclusions.

2. Background

WMA technology was developed to place the asphalt mixture at a lower temperature compared to the conventional mixture, and, in recent years, WMA has gained prominence over conventional HMA approaches due to its sustainability factor. Nevertheless, due to the lower production temperature of WMA, the characteristics of probable moisture damage are more essential to be considered [7], [8]. In this sense, it is necessary to understand the effects occurring at the interface of the mineral aggregate and the asphalt binder. As per the observation of the National Center for Asphalt Technology (NCAT), the reduced mixing temperature used in the preparation of WMA leads to improper drying of the aggregates, and less aging takes place. Figure 1 shows the schematic diagram of moisture entrapment at the asphalt/aggregate interface during WMA production. Digital image processing has become a powerful tool to accurately accomplish the assessment of this type of damage [11].

Generally, image processing experiments are divided into destructive and nondestructive evaluations. In the case of destructive testing, experiments are ought to be carried out on specimen's failure to assess the performance of the material under different conditions by breaking it, whereas in nondestructive testing, the sample is kept intact during its analysis using digital image processing. In the work carried out by Li et al. [17], an artificial neural network was used to detect the cracks in the pavement whereby classification and preprocessing step was performed, followed by Gaussian filter utilization to smoothen the background. At the final step, histogram transformation was applied to highlight the region of crack. The results obtained by image processing were compared with the experimental parameters, which were obtained in lab, and an accuracy of approximately 80% was achieved [17]. Previous studies dealt with the modeling of the microstructure of the asphalt mixture also included sensitivity exploration of aggregate size inside sand mastic [18]. Image analysis is also used for the characterization of air void distribution in asphalt mixes utilizing X-ray Computed Tomography [19, 20]. Moreover, the 3D microstructure of the asphalt concrete can also be reconstructed from pieces of 2D X-ray computed tomography graphics to obtain the required macroscopic specification [21, 22].

Numerous studies and analysis present how to build up and enhance the image processing technique research-based commercial applications such as ImageJ, Image Processing, and Analysis (IPas) or Photo Pro Plus [20, 23, 24] to be implemented on specific domains. Generally, the sizes of modern image collections are large (terabytes and petabytes of data); such vast collections of data cannot be stored and processed efficiently on a single machine. In addition, current image processing algorithms are becoming very challenging and, hence, computationally intensive. There are
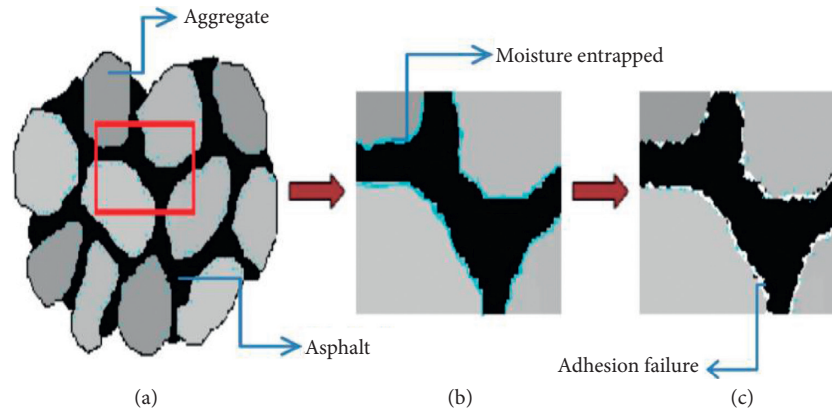
Figure 1: Moisture damage in warm mix asphalt holding moist aggregates (adopted from [16]) (a) Asphalt mixture. (b) Binder aggregate interface. (c) Damaged interface.

many compute intensive steps involved in processing large and complex image datasets to derive modified products, and several efforts are required towards integration of high-performance computing models with image processing algorithms. Although we can process the images batch-wise in these single-processor systems, there will be problems with such processing due to limited capabilities. Henceforth, a parallel programming paradigm for high-end image processing has become deemed necessary.

In image processing, the input image undergoes from several modifications [25, 26]. Implementation of image processing performs four steps: (i) image acquisition, (ii) image enhancement, (iii) image restoration, and (iv) multiresolution processing. In order for an image to be segmented and examined, initially it has to undergo a thresholding process. In image processing of asphalt sample, the image sample undergoes from several preprocessing methods focused on image smoothening to remove the noise from the image [27]. For the analysis of asphalt samples, acquisition of high-quality image is a necessary parameter and so is its intensive computation which is considerably time-consuming using sequential processing. In order to determine ROI in asphalt image sample, initially, the sample image has to be smoothened to segregate the object that determines fractured surface and the remaining unwanted background. For example, to implement this process of segregation, several image processing functions were applied by Dong et al. [28] to analyse the cracks present in asphalt samples. These image processing functions comprised of implementation of filter (i.e., Kalman, Gaussian, Sobel, polynomial, bilateral filter) to remove the noise and then the implementation of thresholding process with Canny edge detection [2] was done followed by morphological implementation operations to analyse contours. Thresholding on grayscale images were also applied to estimate stripping in untreated and treated asphalt mixture [2]. There are limitations of simple thresholding as it is a manual process and it does not resonate well with the dark color aggregate sample. Moreover, simple thresholding does not classify the shadow area accurately [29]. In order to overcome these shortcomings, Lantieri et al. [30], proposed a method of color space conversion, where they converted their sample image from Red-Green-Blue (RGB) color space to YUV color space (Y denotes luminance component, U and V are the chrominance color component).

The conventional image segmentation algorithm primarily contains the segmentation method depending on threshold value, the segmentation method based on the border, and the segmentation method based on the region [31, 32]. However, in order to determine the dominance of any pattern in the image, the k means clustering has turned out to be an efficient method compared to other clustering method as highlighted in Section 1. Cluster analysis is an important aspect for pattern/behavioral analysis [33]. Using k means algorithm, one can find out how to distinguish different kinds of elements by continuously enhancing the subliminal clustering pattern. Different clustering methods are continuously proposed and enhanced with the help of k means algorithm which are widely being used in medical image analysis using both CPU and graphical processing unit (GPU) computing [34]. The clustering process in k means algorithm is time-consuming and increases substantially with the increase in dataset size if implemented sequentially. However, due to the high efficiency of k means algorithm, it is widely utilized at the clustering of large-scale data using GPU [35, 36]. At present, many algorithms are extended and improved by keeping k means algorithm at the pivot. Compared with the traditional k means procedure, the adaptive k means algorithm used by Zheng et al. [29] transforms an image to the LAB color space before segmentation and places the luminance (l) into an adjusted value to reduce the disturbance resulting from the background. In one of the recent studies, sequential k means clustering and machine learning-based classification was used to estimate the stripping in asphalt coating [2]. It is essential to note that with respect to machine learning algorithm, the training process could be time-consuming. However, machine learning-based classification could be helpful to predict the stripping in asphalt mixture sample. It could be inferred from previous research that k means algorithm has the capability to be executed in parallel to cut down the execution time [37, 38]; therefore, a parallel implementation of k means algorithm can considerably enhance the execution time of image segmentation and

additionally optimize the algorithm structure to a specific extent on multiple cores. In order to estimate the average count of pixels associated to region of interest (ROI) on multiple image dataset, Akhtar et al. [39] implemented parallel image segmentation using Map-Reduce framework. In addition to this, Akhtar et al. [12], also made detailed analysis of execution pattern with respect to CPU run time and accuracy for the multiple input image dataset to be segmented in batch. It was evident from their results that higher-order dataset size scaled well with increasing number of compute nodes or compute cores.

In contrast to supervised learning, clustering is seen as an unsupervised learning method, as we do not have the ground truth data in comparing the output of the cluster algorithm with real labels to assess its performance. Numerous k means-based clustering algorithms use distance measurements to determine the similarity between data points [40–42]; however, it is advisable to normalize the data to have a zero average and standard deviation, as the characteristics in each data set would almost have different measurements. The k means algorithm is good at capturing data structures if clusters have a spherical shape [43, 44]. Moreover, in the field of artificial intelligence, k means clustering is used for hard clustering problems. Before discussing the methodology with respect to the k means clustering, Sections 3.1 and 3.2 give a brief description regarding WMA sample preparation and moisture conditioning.

3. Methodology

3.1. Asphalt Mixture Preparation.
Conventional asphalt binder PG-76 was utilized in this study. In Table 1, the rheological properties of the PG-76 binder is summarized. Aggregate-type granite provided by Kuad Quarry Sdn. Bhd., Penang, was utilized in the preparation of the mixtures. The midrange aggregate gradation, type AC 14, used was in agreement with the Malaysian PWD gradation determinations [45, 46] for asphalt concrete. To evaluate the impacts of filler as anti-stripping agent in asphalt mixture, conventional Ordinary Portland Cement (OPC) and Pavement Modifier (PMD) were incorporated. PMD is a greyish-black powder mineral filler created in Malaysia which is utilized as anti-stripping agent. An incorporation of 5% of PMD by total weight of mixture, act as mineral filler in mixtures prepared [27].

Along with two HMA mixtures (with filler types OPC and PMD), WMA mixtures were prepared at different temperatures. The test specimens were compacted to the desired dimensions (height 63.5 mm and diameter 100 mm) by using the Servopac Gyratory Compactor. Air voids of the specimens for moisture sensitivity evaluation were kept at $7 \pm 1\%$. All the specimens were prepared according to the procedures defined by Asphalt Institute Manual (MS-2) [47].

In this study, the HMA mixtures taken as reference material were prepared at 180 degree Celsius and compacted at 170 degree Celsius. However, the WMA specimens were produced at a temperature of up to 30 degree Celsius lower than HMA as shown in Table 2 along with mixture designations.

The optimum binder content was determined as per the procedure of Marshall Mix design ASTM D1559 [48] which follows the the Malaysian Public Works Department (JKR) guidelines [46] for mix type AC-14. Moreover, a gyratory compactor was utilized corresponding to an assumed 30,000,000 equivalent single axle load [27]. Number of gyrations corresponding to initial compaction ($N_{initial}$), design compaction (N_{design}), and maximum compaction (N_{max}) was respectively 8, 100, and 125. In mixture design, the target air voids of (4 ± 0.1) % were used for all the specimens. The designed optimum binder content for both HMA and WMA were obtained as 5.2%. Moreover, for WMA, Cecabase® was used as warm mix additive at an application rate of 0.3% by mass of binder.

3.2. Moisture Conditioning and Assessment.
On compacted asphalt specimens, moisture conditioning was performed to assess the effect of accelerated water conditioning followed by freeze-thaw cycles. For all compacted specimens, the conditioning was carried out as per the guidelines of ASTM D4867 [49]. The only exception in the guideline was the usage of distilled water with an addition of sodium carbonate at 6.662 grams concentration. This addition was used to make the pH level high in order to enhance the stripping rate/damage inside asphalt specimens [11]. The specimens were immersed in the solution and vacuumed for 15 minutes to achieve saturation levels between 55% and 80%. Afterwards, these specimens were exposed to freezing condition at (-18 ± 3) degree Celsius for 16 hours and thawing at 60 degree Celsius for 24 hours as one cycle according to ASTM D4867 [7, 49]. Three sets of specimens, unconditioned dry, conditioned after one freeze-thaw cycle (1FT), and conditioned after thee freeze thaw cycles (3FT), were separated.

The prepared specimens were then subjected to moisture sensitivity test according to American Society for Testing Materials (ASTM D4867) [7, 11, 45, 49]. The indirect tensile strength (ITS) test was used to assess the moisture susceptibility of mixtures at a test temperature of 20 degree Celsius as per the guidelines of ASTM, 2006.

After WMA sample preparation and moisture conditioning assessment, the next step was to develop a parallel image processing-based methodology to determine the adhesion failure in the WMA sample in the lowest possible execution time. For the proposed experiment, we have developed a parallel k means for image processing (PKIP) algorithm to perform the clustering on the WMA sample. Sections 3.3, 3.4, and 3.5 will illustrate the feature space, linearization model, and the implementation of the PKIP algorithm.

3.3. Features of the Input Image (Features Space).
In image processing, a kernel function is used as a linear classifier to solve a nonlinear problem. For the classification of the kernel function-based algorithm, input space comprises of the original image, and feature space comprises the features of corresponding input image. The objective of applying the kernel-based k means clustering on the WMA specimens to distinguish between adhesion failure region and non-

TABLE 1: Properties of binder PG-76.

Property	Penetration at 25 degree celsius, 100 grams, 5 seconds (0.1 mm)	Softening point (celsius)	Ductility at 25 degree celsius (cm)	Flash and fire point (celsius)	Solubility (%)	Specific gravity
Test method	ASTM D5	ASTM D36	ASTM D113	ASTM D92	ASTM D2042	ASTM D70
PG 76	50	69	90	344	99.50	1.02

TABLE 2: Mixing and compaction temperature of asphalt mixtures.

Mixture type	Filler	Mixing temperature	Compaction temperature	Designation
HMA	PMD	180	170	HP180
	OPC	180	170	HO180
WMA	PMD	170	160	WP170
	OPC	170	160	WO170
	PMD	160	150	WP160
	OPC	160	150	WO160
	PMD	150	140	WP150
	OPC	150	140	WO150

adhesion failure region is to bring down the value of its expected square distance d (x) of the point from its center of cluster:

$$d(x) = \frac{1}{Q} \sum_{i=1}^{k} \sum_{p=1}^{n} x_p - c_i^2, \quad (1)$$

where $d(x)$ is the distance from the center of the cluster, Q is the site of the training set, k is the number of clusters, x_p is the current pattern, and C_i is the cluster center within the cluster i.

Now features of the input asphalt specimen $(\phi(x))$ could be computed from input space $X \in (x_1, x_2, x_3, \ldots, x_n)$ using

$$\phi(x) = (\phi_1(x), \phi_2(x), \phi_n(x)). \quad (2)$$

At the ground reality, the value of X is unknown. However, the inner product is known as kernel function (k) shown below:

$$k(x, y) = \langle \phi(x) \cdot \phi(y) \rangle, \quad (3)$$

where $\phi(x)$ and $\phi(y)$ represents the feature space in x and y direction.

3.4. Apply Kernel Function to Linearize the Model.
Assuming in cluster C_p, the average features space $\phi(x)$ could be rewritten as

$$\phi_{avg}(X) = \frac{1}{C_p} \sum_{i=1}^{C_p} \phi(x_i). \quad (4)$$

Now there may exist a point which lies at the outer segment of $\phi(X)$ by appearing as an image to the input space.

In that condition $\phi(X)$ becomes inaccessible, but it is possible to compute its norm.

$$\phi_{avg}(X)^2 = (\phi_{avg}(X_i), \phi_{avg}(X_j))$$

$$= \frac{1}{C_i^2} \sum_{i,j=1}^{C_i} (\phi(x_i), \phi(x_j)) = \frac{1}{C_i^2} \sum_{i,j=1}^{C_i} k, \quad (5)$$

where $k = \phi(x_i), \phi(x_j)$. Now, assuming that, $\phi_{avg}(x) = \tau$. Therefore,

$$\tau^2 = \frac{1}{C_i^2} \sum_{i,j=1}^{c_i} k. \quad (6)$$

Then the distance d_{ij} when the input space is mapped to the features space could be computed as

$$\phi(x) - \tau^2 = d_{ij}^2$$

$$= (\phi(x_i), \phi(x_j)) - 2\tau\phi(x) + (\tau \cdot \tau) \quad (7)$$

$$= k(x_i, x_j) - 2\tau\phi(x) + (\tau_i, \tau_j).$$

It is possible to kernelize the function in equation (7) by using a median filter given by the function: $k'(m, n) =$ median $\{x[i, j], (i, j) \varepsilon W\}$ (median filter function) where W represents a neighborhood defined by the user which is centered around location (m, n) in the image and $x[i, j]$ is 2D array of pixels comprising of ith row and jth column. In addition to the aforementioned filters, there are other filters as well, i.e., Gaussian filter, Bilateral blur filter, and Sigmoid filter which may be used as per the applications requirement [50, 51]. For the proposed experiment, the median filter has been used to kernelize the equation (7).

3.5. Implementation of PKIP Clustering Algorithm.
This section intends to highlight the execution of a streamlined sequential and multicore CPU variant of the k means clustering algorithm. Therefore, the proposed algorithm was customized using multiprocessing programming to obtain comparative outcomes between parallel and sequential execution. The algorithm was tried with different asphalt

mixture image dataset. The image datasets were acquired using a Nikon D800 model. The dimensions of the image samples were 5520 pixels (length) and 3680 pixelsl (width), and the color space of each sample was maintained in RGB prethresholding process. It is deemed necessary to detail out the system configuration in terms of number of logical cores, threading mechanism, and memory on which the PKIP algorithm has to be implemented. The work relied on OS X 10.1 64-bit operating system with a dual-core i5 (Hyper-threading support) at 1.8 GHz with a turbo boost of up to 2.9 GHz having 3 MB shared L3 cache and 8 GB RAM. The proposed PKIP implementation was tested using 4 threads. The version of the C++ compiler used was 4.6.4, while the version of OpenCV (image processing library) used was 3.4.2.

A clear sequential execution was initially performed using the algorithm shown in Pseudocode 1; however, this could not be viewed as an ideal execution on the grounds that the algorithm had a few matrix-like relations which were basic for getting better execution. The matrix computation was involved in the kernelization phase and in the distance computation of each pattern with respect to center of cluster as shown in the Pseudocode 1. In order to get substantial speedup, iterative matrix multiplications could be implemented in a computer environment where it could be executed in parallel.

In the proposed algorithm shown in Pseudocode 1, phase no. 5 holds the key to our algorithm. Phase number 5 could be categorized into two stages, stage I and stage II, using which substantial speedup is obtained for the proposed algorithm. Stage I involves the kernelization phase, and stage II involves an iterative procedure to compute the distance of every pattern to the different center of clusters. Stage I includes registering of the kernel matrix, which uses the kernel function that is connected for each matching pattern. This calculation is similar to matrix multiplication with the exception that for the proposed experiment, several task operations are involved rather than basic matrix multiplication. Moreover, these task operations require dependencies on the utilized kernel functions comprising of subtraction, squaring, and division of the median filter kernel function (Pseudocode 1).

The distance computation between each pattern with respect to the cluster center comprises of three terms as

$$
\text{distance } (i, j) = k'k(x, x) - \frac{2k'}{c_i} \sum_{m=1}^{n} L_{mj} \cdot \tau\phi(x)
$$

$$
+ \frac{k'}{c_i^2} \sum_{m=1}^{N} \sum_{n=1}^{N} L_{mj}L_{nj} \times (\tau_i\tau_j).
$$

(8)

For equation (8), the first term is computed in Stage I. The computation of Stage I comprises of Phase no. (1) to Phase no. (4). The second and third term were computed in stage II. The second term is a direct multiplication of matrices, i.e., L and $\phi(x)$ followed by multiplication by $(2k'/c_i)$. The third term in the above equation depends upon the number of cluster and is continuously computed for every

cluster. Since the third term is an iterative procedure and requires immense matrix multiplication, it was parallelized using multiprocessing programming. As a result, for equation (8), serial execution was exploited to compute the first term, whereas the second and the third term was implemented parallely using multiprocessing as it involved higher degree of iterative matrix multiplication.

In order to assess the performance of the PKIP algorithm for the proposed experiment, we have implemented it using C++ programming language by importing Open Computer Vision (OpenCV) library, which is used for image processing functions. The sequential execution of k means clustering can be further parallelized on multicore CPU. There are numerous application programming interfaces (API) available for parallel processing; however, for the proposed experiment, we have used Open Multi-Processing (OpenMP) for parallelizing the execution. OpenMP is an API for multiprocessing programming which supports shared memory architecture [52]. The prime feature of OpenMP is its vast instruction set i.e., OpenMP pragmas which is used for auto parallelization. For the proposed experiment, OpenMP pragma *omp for* was used to parallelize the intensive matrix computation involved in stage II. In order to divide the execution of Stage I between multiple cores, OpenMP *shared* construct was used.

3.6. Parallelization of K-Means Using Map-Reduce.

In order to parallelize the k means clustering algorithm using Map-Reduce program, Hadoop framework with Hadoop Interface for Image Processing (HIPI) was used. Hadoop is an Apache open source framework written using java programming language by enabling distributed processing of large datasets across clusters of computer nodes using simple programming models [53]. The environment in which Hadoop works comprises of distributed storage and computation across clusters of computer nodes. Hadoop is designed to scale up from single node to thousands of nodes, of which each node offers local computation and storage.

Maps are the individual tasks which converts input records into an intermediate record. Any given input pair may map towards a minimum of zero or a maximum of n output pair. A Hadoop Map-Reduce structure spawn single map task for each input split was produced by the Input-format for the job. The Reduce function of Hadoop framework is the second phase. Reducer minimizes the set of intermediate values passed by mapper and shares a similar key. The quantity of reduces for the job is set by the client by means of the reduce task function. In general, Reducer executions are passed the job configuration for the job through the job configurable class. The framework then calls the function reduce () for each key-value pair for the gathered inputs.

The Map function operates on every point x on the given image dataset. On a given point x, the squared distance between x and every mean is calculated and subsequently the mean M_i is determined which minimizes this distance. On the basis of these parameters, a key-value pair gets emitted

Input sample
   k = number of cluster $(c_1, c_2, \ldots, c_i)$
   Kernel function: $k(x, y) \leftarrow \phi(x_i), \phi(x_j)$
  Phase no
(1) Apply kernel function to transform the input sample $k' = (1/(k_{\text{size}} \cdot \text{width} \times k_{\text{size}} \cdot \text{height})) \times \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 & 1 & 1 \\ 1 & 1 & 1 & \cdots & 1 & 1 & 1 \\ 1 & 1 & 1 & \cdots & 1 & 1 & 1 \end{bmatrix}$,
(2) Compute distance matrix $d_{i,j}^2 = \|\phi(x) - \tau\|^2$
(3) List pattern to the cluster randomly using label matrix, $L_{i,j}$ $L_{i,j} = f(x) = \begin{cases} 1, & \text{if } L_i \text{ in cluster } j, \\ 0, & \text{otherwise.} \end{cases}$
(4) Change $\leftarrow 0$
(5) Distance computation b/w each pattern w.r.t center of cluster)
  while change $\neq 0$ do
    for $j = 1$ to k do
      compute cluster size c_i
       for $i = 1$ to N do
         compute distance of each pattern from the cluster using:
           $k^T k(x_i, x_j) - 2k'/c_i \sum_{m=1}^{n} L_{mj} \cdot \tau\phi(x) + k'/c_i^2 \sum_{m=1}^{N} \sum_{n=1}^{N} L_{mj}L_{nj} \times (\tau_i\tau_j)$.
       end **for**
    end **for**
    Previous $L = L$
    update label L
    $L = \min. val$ of d along column
    if Previous $L = L$ **then**
      change $= 0$
    end **if**
  end **while**

PSEUDOCODE 1: k-means using sequential execution.

with an index of mean "i" as key and $(x, 1)$ as the value. Therefore, a map function could be framed as

$$k \text{ means map } (x) : \text{compute} \left(\text{argmin}_i \|x - M_i\|^2, (x, 1) \right). \tag{9}$$

On the other hand, the Reduce function is just a pairwise summation performed on the values associated to each key. For instance, if two value pairs $[(x, q), (y, r)]$ are associated to a particular key, then a combined formation is possible by adding all elements in the given pair. In this regard, k means reduce function could be formed as follows:

$$k \text{ means Reduce } (i, [(x, q), (y, r)]) : \text{return } (i, (xx + yy, qq + rr)). \tag{10}$$

Set of k values could be formed using the Map-Reduce characteristic of two functions shown as

$$\left(i, \left(\sum_{x \in P_i} x, |P_i| \right) \right), \tag{11}$$

where P_i denotes the set of points closest to mean M_i. Now the updated means could be computed as

$$\text{Mi} \longleftarrow \frac{1}{|P_i|} \sum_{x \in P_i} x. \tag{12}$$

For the proposed experiment, initial cluster centres (means) were determined using the canopy algorithm [54]

whereby a set of initial distance threshold, i.e., T1 and T2 were assigned for each sample image such that T1 > T2. Post threshold initialization, the mapper function selects a sample vector randomly from the sample image and assigns it as a central vector of the canopy, and accordingly, it traverses the entire sample image. The distance between the scene image and the canopy central vector has to be less than T1 in order to get classified as canopy. If it is less than T2, then it gets discarded from the dataset. Finally, the output of the mapper function gets processed by the reducer to integrate the central vectors of the canopy. This generates a new canopy of the central vector which is considered as the initial clustering center.

In order to calculate the distance between a point x and each of the means using Map function, every compute node must possess set of current values of means. Therefore, in this regard, new means are circulated to all the compute nodes after the completion of each iteration. If the convergence is achieved after a specific iteration, then the execution gets completed; else, the new means/centroid are computed again using iterate () function.

Pseudocode 2 shows the implementation of k means using Map-Reduce.

Our experimental setup consists of 4 machines comprising of 1 master and 3 slaves. The master node is responsible to take the input image files. For the proposed experiment, Hadoop (version 2.7.1) framework was used in order to implement the parallel computation using 4 nodes. Figure 2 shows the setup configuration. The setting of Hadoop parameters was amended as shown in Table 3. Table 4 shows the specs of the nodes.

(1) Choose k initial value for the input image.
(2) Apply Map-Reduce $\forall$ k.
       iterate (point, centroids)
       {Assign points => clustersmap {(point => (point nearest to centroids) => (point, 1), (point n, count n))}reduce {(((point1, count1), (point2, count2)) => (point1 + point2, count1 + count2)}Total_mapValues {(point_Total, count_Total) => point_Total/ count_Total}
(3) Record the new means (Updated Centroid) post Map-Reduce.
(4) Circulate the updated centroid to each node in the cluster.
(5) **If** (converged (centroids, Updated_Centroids))
       Then finish execution
       Else iterate (points, Updated_Centroids)

PSEUDOCODE 2: k means using Map-Reduce.
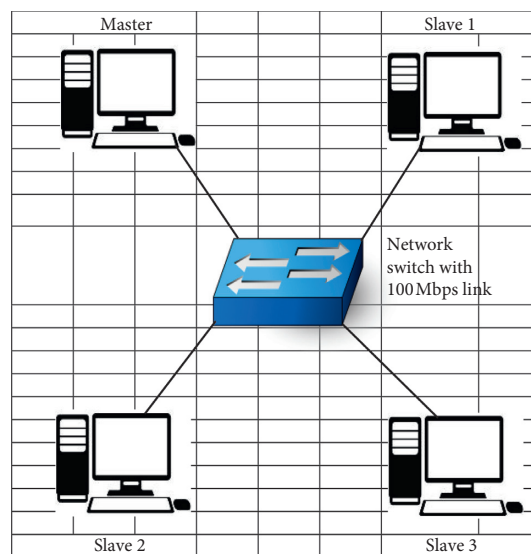
FIGURE 2: Experimental setup of Hadoop nodes.

TABLE 3: Configuration of Hadoop parameters.

Block size (MB)	Number of task mapping per node	Number of task reducing per node	Replication factor	Scheduler configuration	k value
128	2	1	2	First-in-first-out (FIFO)	10

TABLE 4: Configuration of nodes.

Nodes	RAM (GB)	Processor (GHz)	Dedicated hard disk drive (HDD) (GB)	No. of cores	Operating system
Master Slave 1 Slave 2 Slave 3	8	1.8	50	2	OS X 10.1

3.7. Thresholding of Processed Samples.

For the analysis of asphalt mixture specimens post k means clustering, HSV thresholding was used (see Appendix section) [55].

In the proposed experiment, background and foreground segregation was significant to gain clear visibility of the ROI. For instance, the HSV color space for the boundary condition {[0, 1], [0, 1], [0, 255]} belongs to {Hue_{input}, Sat_{input}, Val_{input}}, the region of interest (ROI) being a specific tuned color object. Then the computation of the threshold pairs, i.e., (Hue_{lower}, Hue_{upper}), (Sat_{lower}, Sat_{upper}), and (Val_{lower}, Val_{upper}) could be utilized to convert the HSV color space image to the binary form using the below equation:

$$C(x, y) = \begin{cases} 1, Hue_{lower} < Hue_{input}(x, y) < Hue_{upper}, \\ Sat_{lower} < Sat_{input}(x, y) < Sat_{upper}, \\ Val_{lower} \leq Val_{input}(x, y) \leq Val_{upper}, \\ 0, \text{Otherwise.} \end{cases}$$

(13)

Here $C(x, y)$ is the segmented part. The equation (13) illustrates that if the HSV values for the pixels of the input image lies within the range of lower bound to upper bound values, then its associated output pixel belongs to class object 1, otherwise it gets designated to null (0) [56].

The flowchart in Figure 3 illustrates the HSV thresholding along with computation of zero and nonzero pixels. Figure 4 shows different images of specimens from the WMA dataset (Dry, 1FT, and 3FT) and Figure 5 shows its raw RGB model whereby it could be observed that the variable intensities of the brown component represents the adhesion failure in WMA samples. Figure 6 shows the different images of specimens from the HMA dataset and Figure 7 shows its associated raw RGB model. From the visual assessment point of view, the white region within the sample represents broken aggregates. If the RGB model is observed minutely for WMA and HMA samples, then a slight difference could be observed with respect to the brown component pattern for WMA and HMA samples, i.e., higher degree of the brown component is visible for HMA samples if compared to WMA for all the three categories, i.e., Dry, 1FT, and 3FT. (Figures 4–7).

4. Results and Discussion

The proposed work highlights the efficiency of PKIP algorithm for CPU-based execution by testing several datasets of asphalt mixtures. Analysis was done for the performance of proposed algorithms in terms of execution time and accuracy. For this study, three categories of dataset were evaluated, namely, dry, 1FT, and 3FT. At the same time, the influence of the mixing temperature (i.e., 180°C for HMA and 170°C, 160°C and 150°C for WMA) and type of filler (OPC and PMD) were analysed as well. The details of the different datasets are shown in Table 5.

In order to remove the noise, median filter was applied, and subsequently, the images of WMA along with its RGB color model and HMA samples along with its RGB color model is shown in Figures 8, 9, 10, and 11. After having a clear observation of the filtered images and its RGB color model, it could be seen that substantial noise elements have been blurred and the sample images has been polished. Post the application of median filter, the process of k means clustering becomes fine-tuned and thus helps in providing the accurate results.

Before initiating the explanation of CPU-based performance, it is required to highlight the accuracy of its execution timing result. With respect to execution time, it is important to note that there were fluctuations in the reading after each run of PKIP algorithm for every sample; therefore, the recorded errors were the best-obtained ones post four runs of the proposed algorithm. This performance comparison is shown in Figure 12. The fluctuation in the obtained values was within the range of |5%| from the best recorded value. Moreover, it is necessary to clarify that this error is due to the communication overhead among the threads (virtual version of CPU core) which results is extra waiting time.

As mentioned earlier in Section 3.5, with respect to the PKIP, the algorithm constitutes two stages, i.e., stage I involves the kernelization phase and stage II involves iterative procedure to compute the distance of every pattern to the different center of clusters. As stage II is an iterative process, it has to complete a predefined maximum number of iterations to attain the convergence. With respect to the proposed experiment, for each value of k, the cost of k means function was computed using vegas-viz plot library for Scala (see Appendix section) [57]. The best choice for k was chosen by observing the value which got minimized with little return gain. For the proposed PKIP algorithm, the best segregation was obtained by setting the cluster size to 10. Figure 13 shows one of the specimens (WP170 subjected to 3FT) with varying number of clusters (k). It is observed from each clustering stage that the different intensities of brown color (adhesion failure) are getting normalized. By keeping the value of $k = 10$, a fine segregation is achieved with respect to different intensities of color component which makes it a favorable factor for HSV thresholding. For the proposed experiment, the noise was removed using median filter function. From the visual assessment also, it is observed from Figure 13 that post cluster 5, maximum degree of segregation is achieved, or the intra-cluster variation has reduced within the sample region. However, we have chosen $k = 10$ as per the k value and cost of k means function as the best case.

Figure 14 shows the RGB color model along with histogram in RGB and HSV color space for the sample input image. In the RGB model, the encircled portion represents the adhesion failure. It is important to note that before initiating the clustering process, histogram equalization was applied on input image to maintain the uniform luminosity. It is observed from the histogram that most of the variation in the RGB components corresponds to a uniform lower range value of the pixel (low luminosity). However, at the extreme right of the histogram, higher peaks of RGB is to be seen, which clearly signifies clustered noise with high luminosity. It is also essential to note that the more the luminosity is, the higher the v value in HSV histogram is.

4.1. Discussion on CPU Implementation. In regard to the CPU implementation of the proposed PKIP algorithm, several ideas are noteworthy, especially with respect to the parallel code which leverages on the number of CPU thread(s) which is considered to be a virtual version of CPU core. Figure 15 shows the effect of incrementing the threads to execute PKIP clustering on different asphalt mixture datasets for $k = 10$. As shown in Figure 15, increasing the number of threads from 1 to 2 brings down the computational time in a nonlinear manner. This is due to the fact that all stages of computation involved in equation (8) do not scale directly with the available cores. For equation (8), stage I which does not possess iterative computation showed higher degree of linear scaling compared to stage II. This particularly signifies that all the components of stage II were not completely parallelized, and there was some composition of serial execution remaining in it. It is evident from the
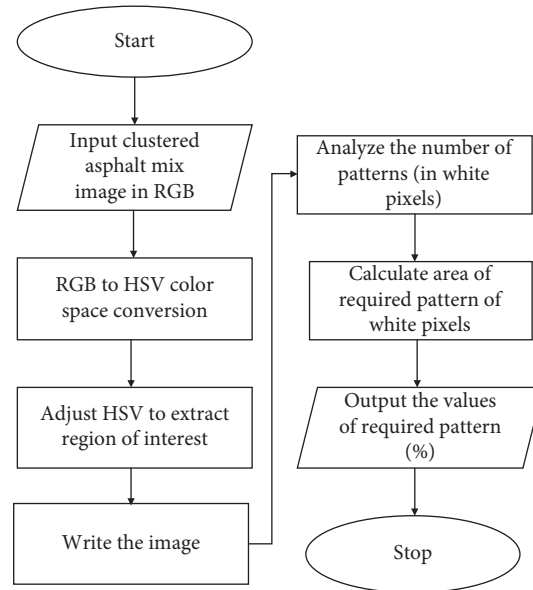
FIGURE 3: Process of image thresholding.

(a)

(b)

(c)
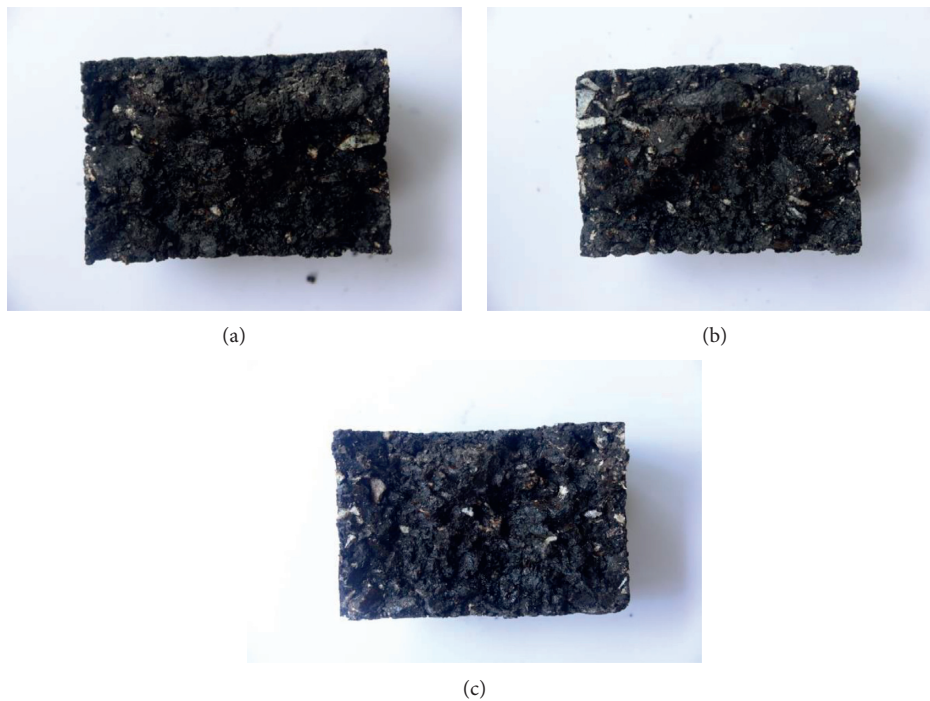
FIGURE 4: Sample specimen of (a) Dry-WP170, (b) 1FT-WO170, (c) 3FT-WP170.

execution time of C program (serial execution) and C + OpenMP (see Appendix section) program (parallel execution) that there is an approximately 25% to 30% improvement in terms of execution time with respect to C + OpenMP program, i.e., parallel execution of k means clustering.

Moreover, when the number of threads is increased from 2 to 3, then it could be observed from the execution timing results that there is not much difference between single thread and triple thread results. In addition to this, when the

number of threads was increased from 3 to 4, then there was an approximately 25% to 30% increment in the execution time of C + OpenMP. Therefore, it is evident from the obtained results that increasing the number of threads does not always increase the execution time subjected to the specification of the CPU. With context to effect of hyperthreading, refer to Appendix section.

Tables 6, 7, and 8 show the execution timing (sec) results of PKIP algorithm implemented both sequentially and parallelly on the WMA and HMA samples. Table 9 shows the

(a)

(b)

(c)
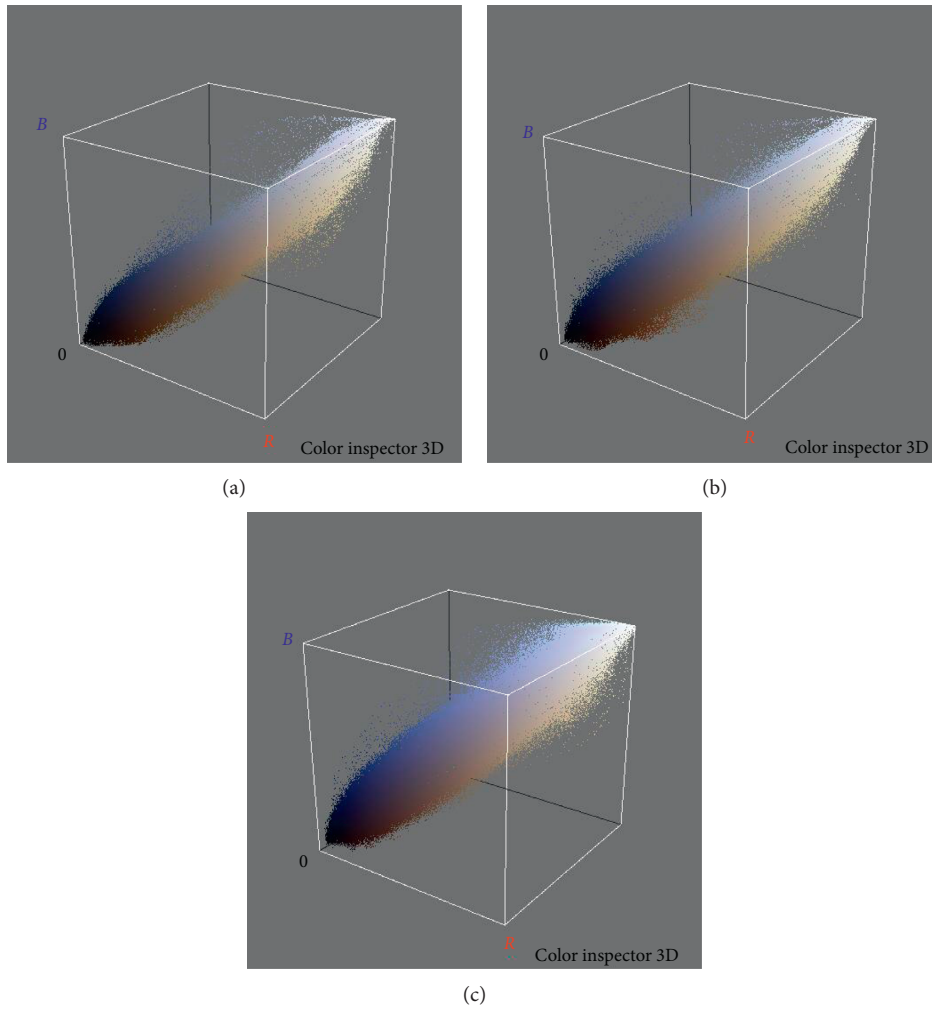
FIGURE 5: RGB-model of (a) Dry-WP170, (b) 1FT-WO170, (c) 3FT-WP170.

(a)

(b)

(c)
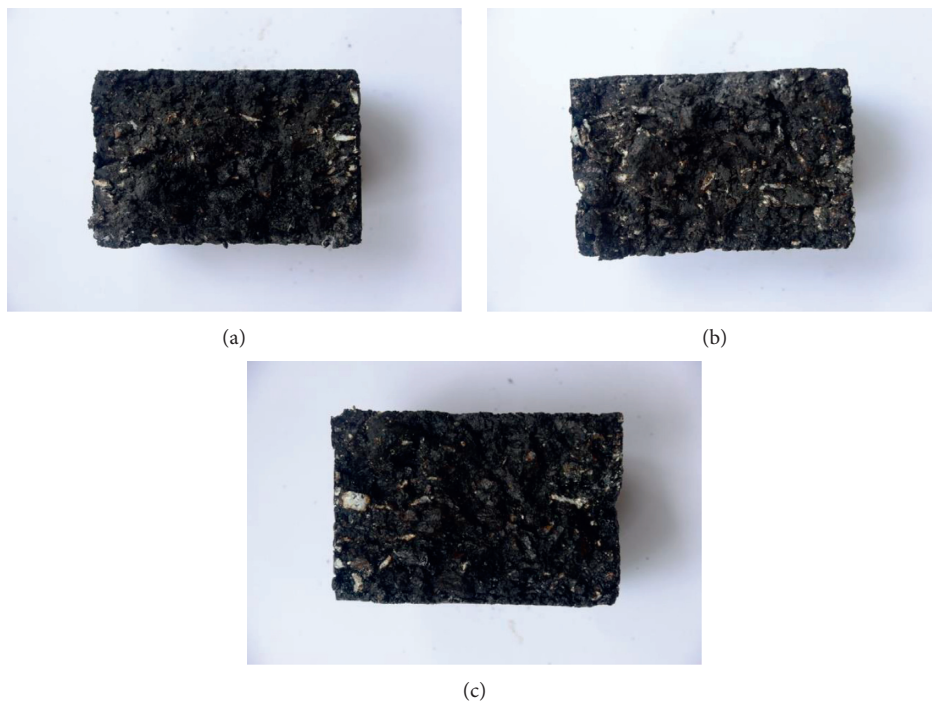
FIGURE 6: Sample specimen of (a) Dry-HP180, (b) 1FT-HO180, (c) 3FT-HP180.

(a)

(b)

(c)
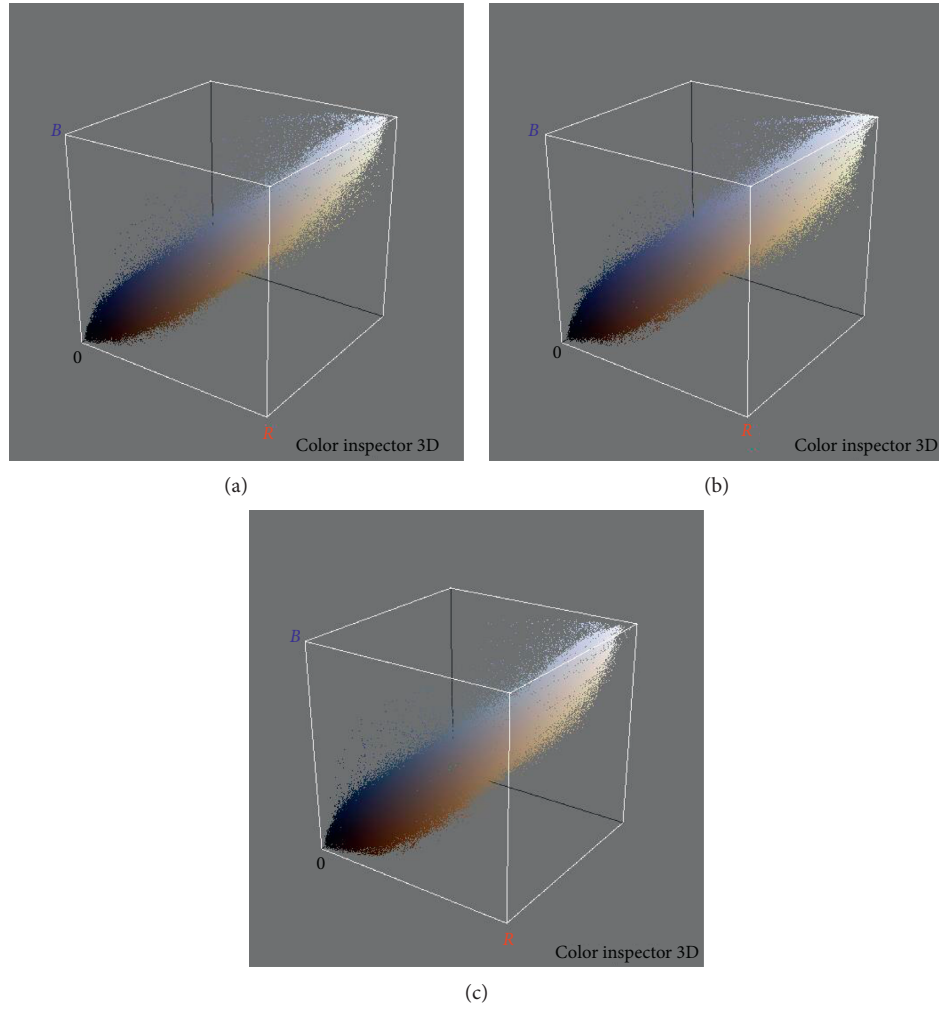
FIGURE 7: RGB-model of (a) Dry-HP180, (b) 1FT-HO180, (c) 3FT-HP180.

TABLE 5: Datasets of asphalt mixture samples.

	Datasets				
Dry		1FT		3FT	
Sample	Sample code	Sample	Sample code	Sample	Sample code
WP170	1	WP170	9	WP170	17
HP180	2	WP160	10	WP150	18
HO180	3	WO170	11	WO170	19
WO150	4	WO150	12	WO150	20
WO160	5	WP160	13	WP160	21
WO170	6	WO160	14	WO160	22
WP160	7	HO180	15	HO180	23
WP150	8	HP180	16	HP180	24

gain in speedup between sequential implementation using C program (Serial Execution) and the parallel implementation using C + OpenMP for $k = 10$.

It is inferred from Tables 6, 7 and 8 that when the number of clusters (k) increases then PKIP algorithm scales well if compared with the serial or sequential programming for all the datasets. In particular, the best timing difference was obtained for $k = 2$, between parallel k means and sequential. It can be observed that for dry conditions, a wider gap by considering lower execution time for C + OpenMP is 9.49 sec, similarly for $k = 4$, 6, 8, and 10, the associated timing difference is 14.84 sec, 22.30 sec, 30.73 sec, and 37.17 sec, respectively. For 1FT condition specimens, the best timing difference for $k = 2$, between parallel k means and sequential k means is 9.43 sec while for $k = 4$, 6, and 10 the associated timing difference is 18.87 sec, 28.43 sec, 38.34 sec, and 47.47 sec, respectively. Finally, for the specimens conditioned with

(a)

(b)

(c)
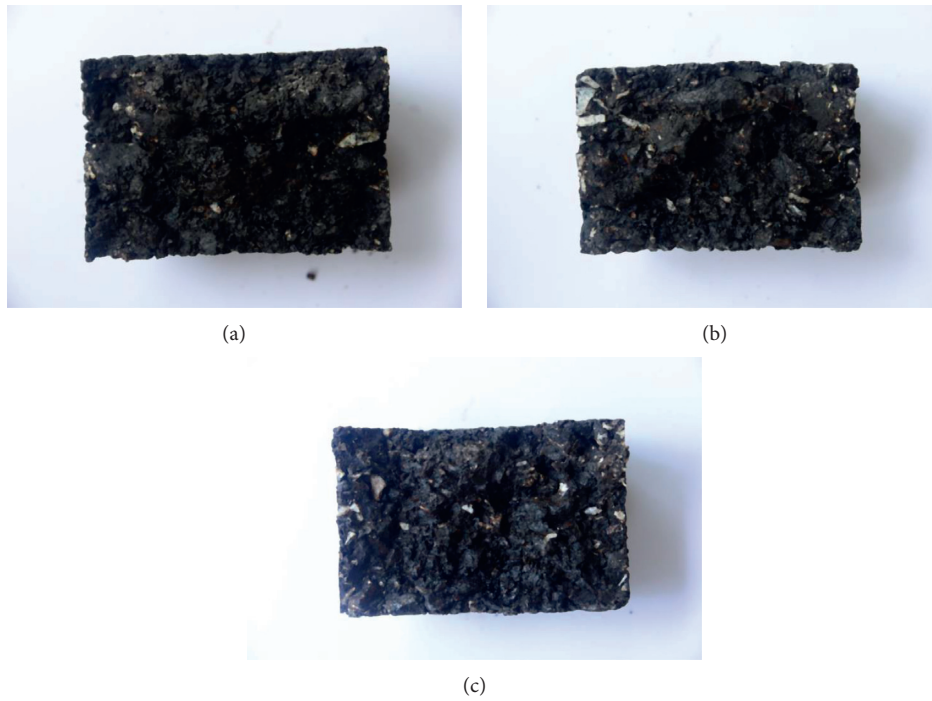
Figure 8: Post median filter-sample specimen of WMA. (a) Dry-WP170, (b) 1FT-WO170, (c) 3FT-WP170.

(a)

(b)

(c)
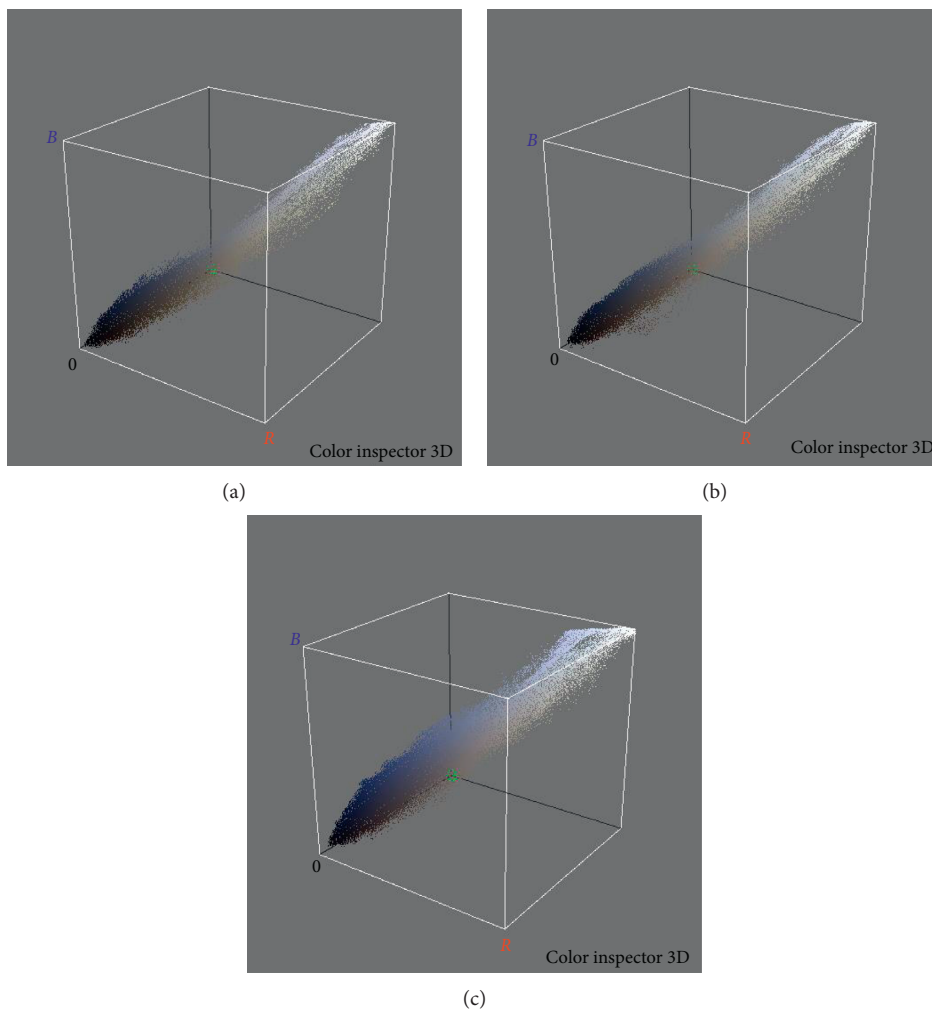
Figure 9: RGB-Model of median filtered. (a) Dry-WP170, (b) 1FT-WO170, (c) 3FT-WP170.

(a)

(b)

(c)
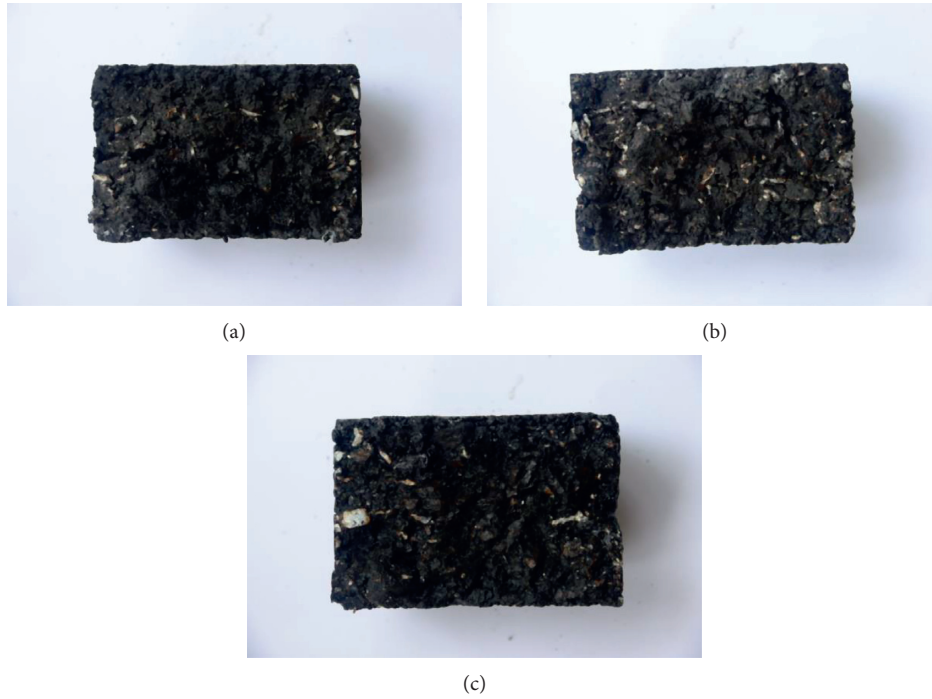
Figure 10: Post median filter-sample specimen of HMA. (a) Dry-HP180, (b) 1FT-HO180, (c) 3FT-HP180.

(a)

(b)

(c)
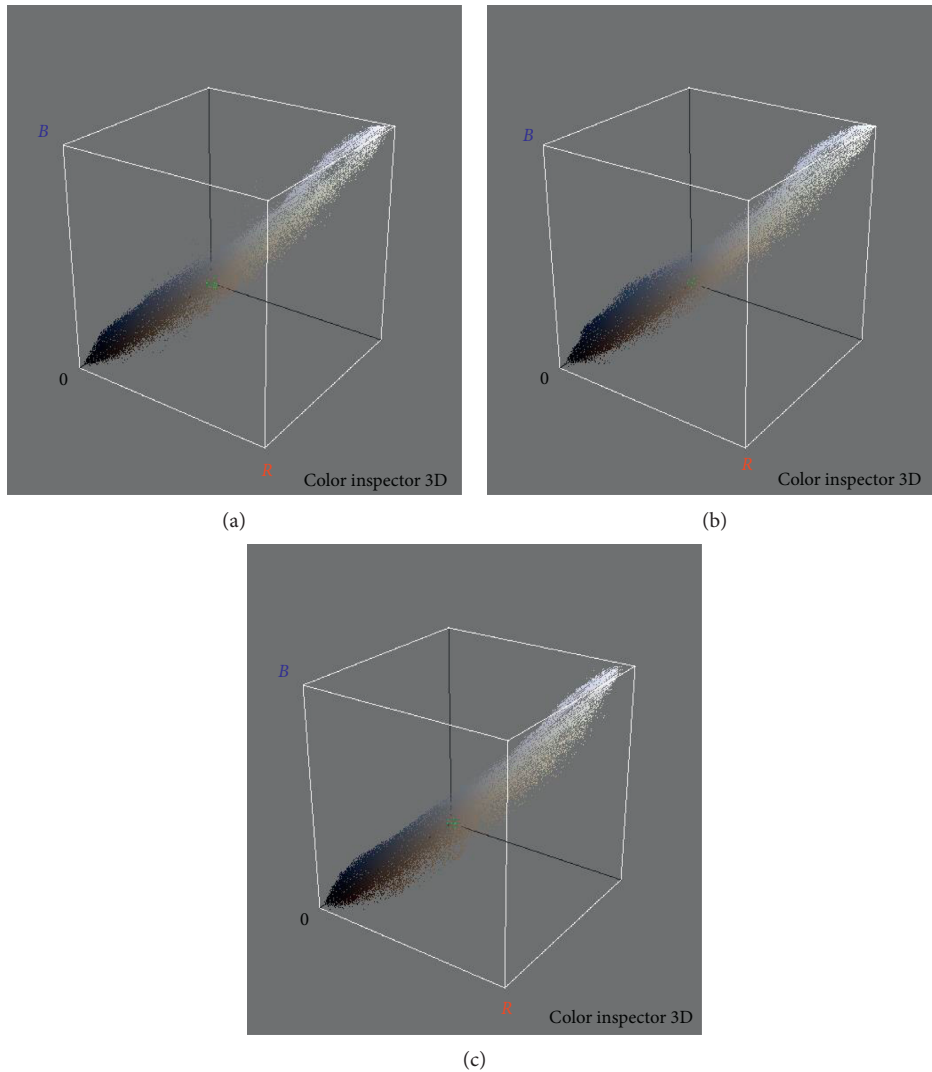
Figure 11: RGB-Model of median filtered. (a) Dry-HP180, (b) 1FT-HO180, (c) 3FT-HP180.
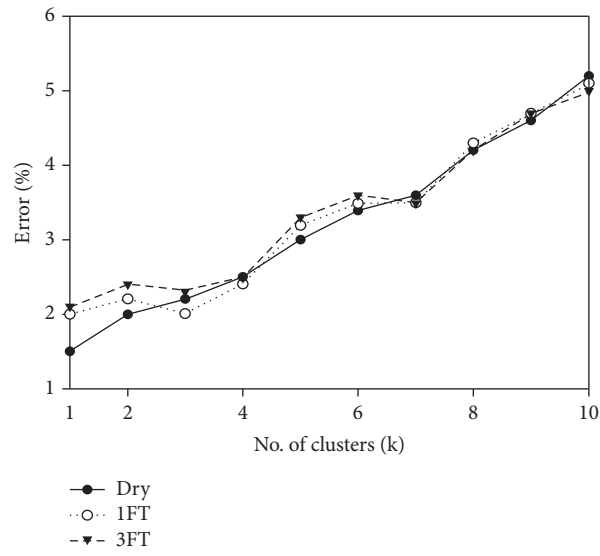
FIGURE 12: Accuracy of PKIP using openMP.

Original

(a)

$k = 2$

(b)

$k = 3$

(c)

$k = 4$

(d)

$k = 5$

(e)

$k = 6$

(f)
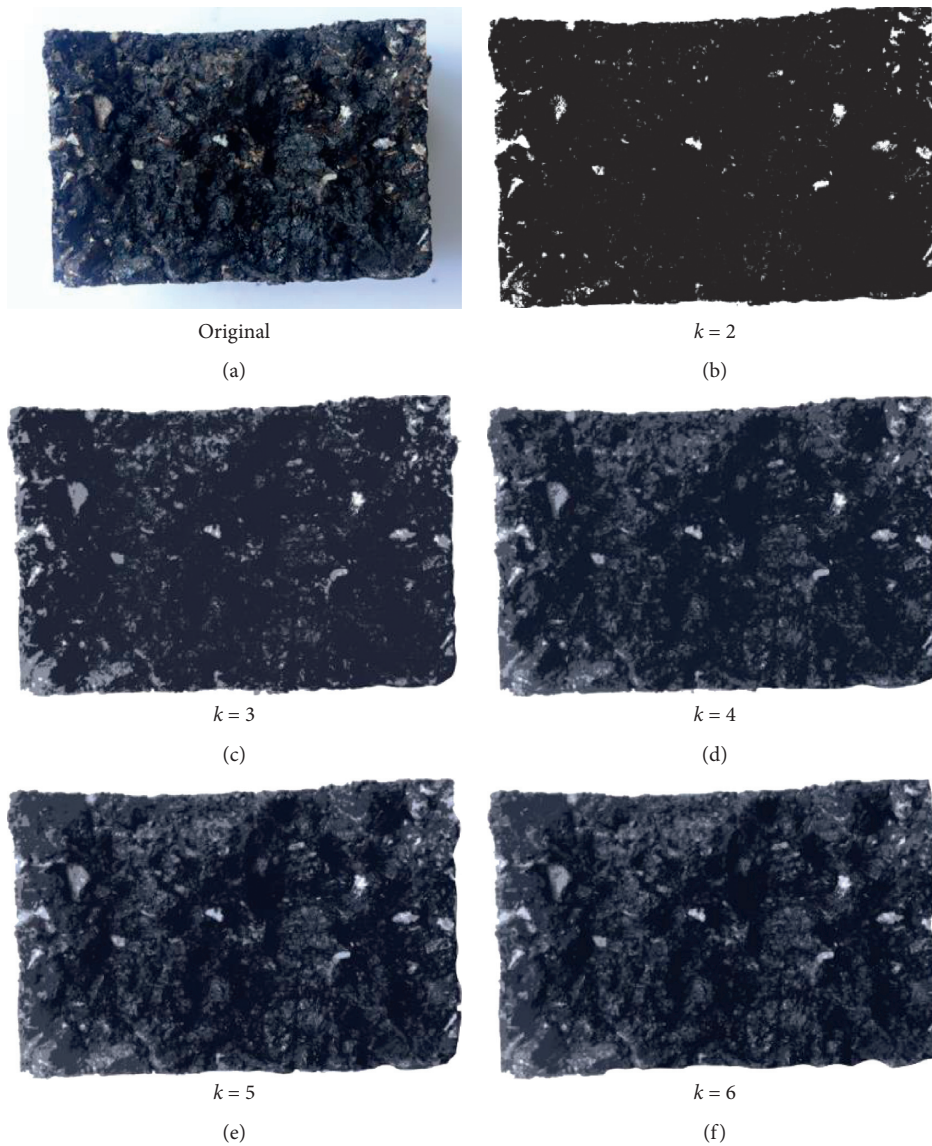
FIGURE 13: Continued.
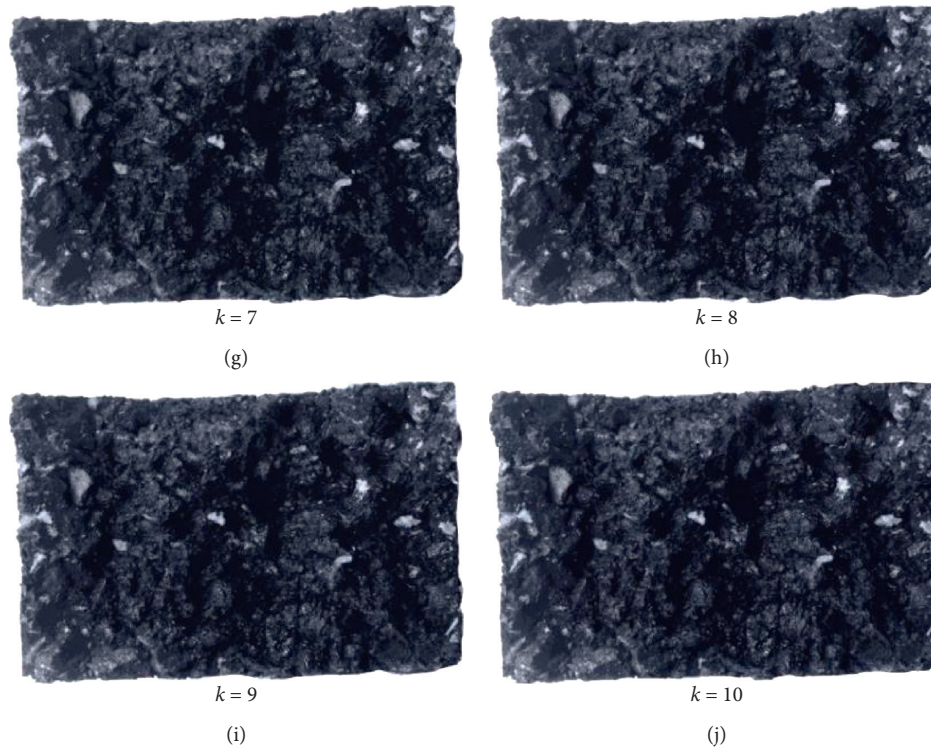
FIGURE 13: Clustered images of WP170 subjected to 3FT sample specimen. (a) Original. (b) $k = 2$. (c) $k = 3$. (d) $k = 4$. (e) $k = 5$. (f) $k = 6$. (g) $k = 7$. (h) $k = 8$. (i) $k = 9$. (j) $k = 10$.

3FT, the best timing difference for $k = 2$, between parallel k means and sequential k means is 9.03 sec, whereas for $k = 4$, 6, 8 and 10 the associated timing difference is 18.07 sec, 22.34 sec, 31.66 sec, and 45.18 sec, respectively.

4.2. Map-Reduce-Based Implementation. Tables 10, 11, and 12 compare the execution timing result of Map-Reduce-based implementation of PKIP algorithm over single node and multiple nodes for Dry, 1FT, and 3FT image datasets. Table 13 shows the speedup comparison of Map-Reduce-based PKIP algorithm implementation with its sequential implementation using C-program and parallel implementation using C + OpenMP. It could be inferred from the results that the Map-Reduce based PKIP algorithm scales well with the increase in nodes. For the execution with respect to single-node implementation of PKIP algorithm, an overhead resulted post splitting of algorithm in Map and Reduce phase. It is also essential to note that Map-Reduce-based PKIP algorithm implementation can outperform C + OpenMP-based PKIP implementation by increasing the number of compute nodes due to the fact that C + OpenMP cannot be implemented over distributed nodes. After performing the initial experiments, we observed that Hadoop block sizes should depend upon the input image dataset. In Hadoop, the data files are divided into blocks. When the block size is too small, the number of data block collaborations involved may increase, thereby affecting the overall results [58].

If we closely observe the execution timing in Tables 10–13, then we can see some similarity between the execution timing of C + OpenMP and single-node implementation of PKIP algorithm for $k = 10$. Moreover, up to 26% reduction could be observed in execution timing for $k = 10$, implemented over 4 nodes if compared to single-node execution timing. However, only up to 4.3% reduction could be observed in execution timing for $k = 4$, implemented over 4 nodes if compared to single node execution timing.

Obtained results illustrate the superiority of the proposed Map-Reduce-based PKIP algorithm when applied to different image datasets. Map-Reduce-based PKIP algorithm enhanced the clustering response time by leveraging on Hadoop framework characteristics. Impact of the number of compute nodes used on the clustering algorithms can also be studied for 3 categories of image dataset by increasing the number of compute nodes (i.e., from 2 to n). Scalability of Map-Reduce based PKIP algorithm can be tested by researchers by maintaining the same type of sample image datasets but varying the cluster size.

With respect to the measure in accuracy of execution timing for PKIP implementation using Map-Reduce, the recorded errors were the best-obtained ones post four runs similar to the case of C + OpenMP implementation. Figure 16 shows the fluctuation or the accuracy in execution time reading for Map-Reduce-based PKIP implementation. For the PKIP implementation using Map-Reduce, the fluctuation in the obtained values were within the range of |1.8%| from the best recorded value. The probable reason for
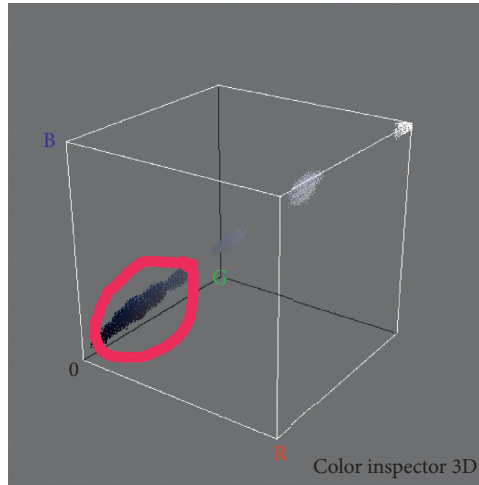
(a)
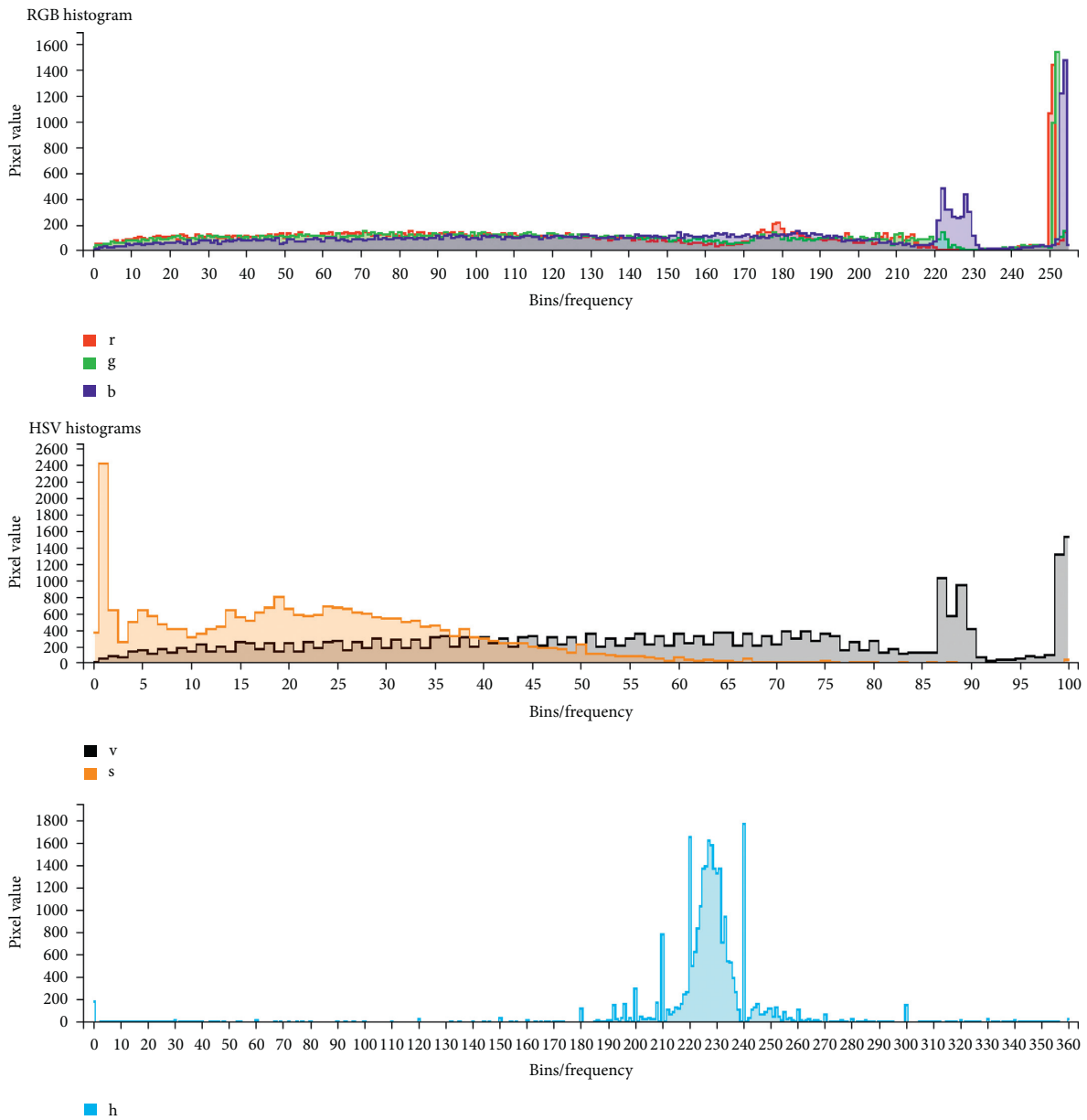
(b)

Figure 14: (a) RGB color model; (b) histogram for RGB and HSV color space.
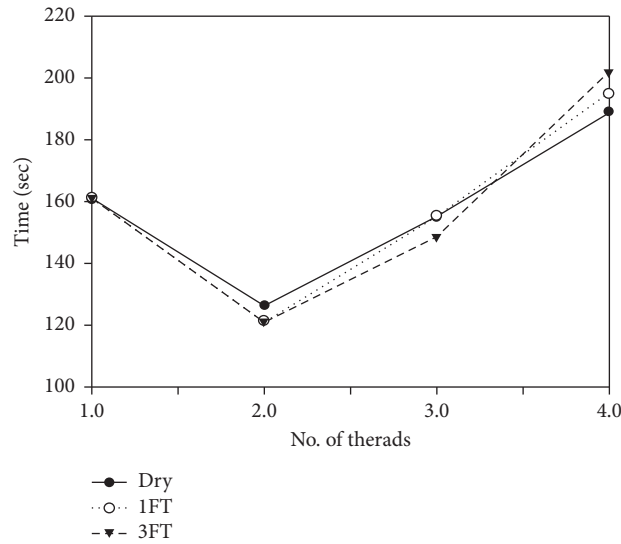
FIGURE 15: Execution time analysis by varying number of threads.

TABLE 6: Serial v/s parallel execution time (sec) for dry condition.

Sample	K = 2		K = 4		K = 6		K = 8		K = 10	
	Serial exec	Parallel exec	Serial exec	Parallel exec	Serial exec	Parallel exec	Serial exec	Parallel exec	Serial exec	Parallel exec
WP170	29.21	22.53	59.43	46.07	87.65	67.60	120.86	93.14	161.08	127.68
HP180	29.23	22.29	59.46	45.59	87.69	66.89	120.92	92.19	161.15	126.49
HO180	29.22	22.08	59.45	45.16	87.67	66.24	120.90	91.32	161.12	125.40
WO150	29.19	22.36	59.38	45.72	87.57	67.08	120.77	92.44	160.96	126.80
WO160	29.37	22.04	59.75	45.08	88.12	66.12	121.50	91.16	161.88	125.20
WO170	28.18	22.36	57.37	45.72	84.55	67.08	116.74	92.44	155.92	126.81
WP160	29.47	22.04	59.94	45.10	88.42	66.12	121.89	91.16	162.37	125.20
WP150	29.24	22.62	59.49	46.24	87.74	67.86	120.99	93.48	161.24	128.10

TABLE 7: Serial v/s parallel execution time (sec) for 1FT condition.

Sample	K = 2		K = 4		K = 6		K = 8		K = 10	
	Serial exec	Parallel exec	Serial exec	Parallel exec	Serial exec	Parallel exec	Serial exec	Parallel exec	Serial exec	Parallel exec
WP170	29.12	19.69	59.25	40.38	87.57	59.07	120.10	81.76	160.62	113.45
HP180	29.36	22.53	59.73	46.07	88.09	63.60	124.46	90.14	161.82	127.67
HO180	29.32	20.39	59.64	41.79	87.96	61.19	122.29	84.58	161.61	116.98
WO150	29.19	21.06	59.39	43.13	87.58	63.20	120.78	87.27	160.98	120.33
WO160	29.26	22.17	59.52	45.35	87.78	66.52	121.04	91.70	161.30	125.88
WO170	29.45	22.32	59.91	45.65	88.37	66.98	121.83	92.31	162.29	126.64
WP160	29.85	22.46	60.70	45.92	89.55	67.38	123.40	92.85	164.26	127.31
WP150	29.23	19.74	59.47	40.48	87.71	59.22	120.95	81.96	161.18	113.71

TABLE 8: Serial v/s parallel execution time (sec) for 3FT condition.

Sample	K = 2		K = 4		K = 6		K = 8		K = 10	
	Serial exec	Parallel exec	Serial exec	Parallel exec	Serial exec	Parallel exec	Serial exec	Parallel exec	Serial exec	Parallel exec
WP170	26.00	19.25	53.01	39.51	78.02	57.76	120.03	80.02	145.04	111.27
HP180	29.14	22.04	59.28	45.09	87.42	66.14	120.57	91.19	160.71	125.23
HO180	29.45	22.01	59.91	45.02	88.37	66.03	121.83	91.05	162.28	125.06
WO150	29.44	20.40	59.88	41.81	88.32	61.22	121.77	84.62	162.21	117.03
WO160	29.47	22.19	59.94	45.39	88.41	66.58	121.89	91.78	162.36	125.98
WO170	29.22	22.28	59.45	45.56	87.68	66.84	120.91	92.12	161.13	126.40
WP160	29.37	21.01	59.75	43.02	88.13	63.04	121.51	87.05	161.89	120.06
WP150	29.46	20.43	59.92	41.87	88.38	61.30	121.84	84.74	162.30	117.17

TABLE 9: Computation of speedup ($k = 10$).

Sample	Execution time using C-Sequential (sec)	Execution time using C + OpenMP (sec)	Speedup
Dry dataset			
WP170	161.087	127.680	1.26
HP180	161.153	126.498	1.27
HO180	161.129	125.403	1.28
WO150	160.964	126.808	1.26
WO160	161.883	125.204	1.29
WO170	155.927	126.810	1.22
WP160	162.370	125.200	1.29
WP150	161.240	128.107	1.25
1FT dataset			
WP170	160.629	113.453	1.41
HP180	161.825	127.679	1.26
HO180	161.616	116.986	1.38
WO150	160.983	120.339	1.33
WO160	161.305	125.882	1.28
WO170	162.292	126.643	1.28
WP160	164.261	127.315	1.29
WP150	161.188	113.712	1.41
3FT dataset			
WP170	145.048	111.279	1.30
HP180	160.713	125.238	1.28
HO180	162.289	125.065	1.29
WO150	162.214	117.035	1.38
WO160	162.365	125.981	1.28
WO170	161.139	126.401	1.27
WP160	161.896	120.068	1.34
WP150	162.309	117.178	1.38

TABLE 10: Single node v/s multiple Hadoop nodes execution time (sec) for dry condition.

Sample	$K = 4$		$K = 10$	
	Single node	4 nodes	Single node	4 nodes
WP170	69.96	67.02	129.59	103.67
HP180	69.23	66.26	128.38	98.85
HO180	68.55	65.59	127.28	99.27
WO150	69.42	66.44	128.70	102.96
WO160	68.43	65.49	127.07	98.81
WO170	69.42	66.51	128.71	98.85
WP160	68.43	65.55	127.07	96.45
WP150	70.23	67.28	130.02	101.01

TABLE 11: Single node v/s multiple Hadoop nodes execution time (sec) for 1FT condition.

Sample	$K = 4$		$K = 10$	
	Single node	4 nodes	Single node	4 nodes
WP170	61.13	58.56	115.15	86.12
HP180	65.82	63.06	129.58	98.19
HO180	63.33	60.70	118.73	92.13
WO150	65.41	62.66	122.13	95.27
WO160	68.84	65.95	127.76	99.33
WO170	69.32	66.41	128.53	98.55
WP160	69.73	66.80	129.21	103.37
WP150	61.29	58.71	115.41	86.30

this fluctuation is due to the communication cost of mapping and reducing. Moreover, in this regard, researchers are free to explore the possibility of reducing this fluctuation by optimized scaling and tight scheduling of Hadoop parameters. It is also to be observed that the fluctuation in the execution time reading is lower if compared to C + OpenMP readings. This shows the stability of Hadoop's Map Reduce framework when scaling is taken into account.

TABLE 12: Single node v/s multiple Hadoop nodes execution time (sec) for 3FT condition.

Sample	K = 4		K = 10	
	Single node	4 nodes	Single node	4 nodes
WP170	59.78	57.27	112.93	83.57
HP180	68.45	65.57	127.10	94.911
HO180	68.34	65.39	126.93	95.21
WO150	63.36	60.64	118.78	89.23
WO160	68.91	65.98	127.86	96.19
WO170	69.17	66.27	128.29	94.93
WP160	65.24	62.50	121.86	90.59
WP150	63.44	60.78	118.92	95.14

TABLE 13: Computation of speedup (k = 10) for multiple Hadoop nodes v/s C-sequential & C + OpenMP.

Sample	Execution time using C-Sequential (sec)	Execution time using C + OpenMP (sec)	Execution time using 4 nodes (sec)	Speedup (C + OpenMP/4 nodes)	Speedup (sequential-C/4 nodes)
Dry dataset					
WP170	161.08	127.68	103.67	1.23	1.55
HP180	161.15	126.49	98.85	1.27	1.63
HO180	161.12	125.4	99.27	1.26	1.62
WO150	160.96	126.8	102.96	1.23	1.56
WO160	161.88	125.2	98.81	1.26	1.63
WO170	155.92	126.81	98.85	1.28	1.57
WP160	162.37	125.2	96.45	1.29	1.68
WP150	161.24	128.1	101.01	1.26	1.59
1FT dataset					
WP170	160.62	113.45	86.12	1.31	1.86
HP180	161.82	127.67	98.19	1.30	1.64
HO180	161.61	116.98	92.13	1.26	1.75
WO150	160.98	120.33	95.27	1.26	1.68
WO160	161.30	125.88	99.33	1.26	1.62
WO170	162.29	126.64	98.55	1.28	1.64
WP160	164.26	127.31	103.37	1.23	1.58
WP150	161.18	113.71	86.30	1.31	1.86
3FT dataset					
WP170	145.04	111.27	83.57	1.33	1.73
HP180	160.71	125.23	94.911	1.31	1.69
HO180	162.28	125.06	95.21	1.31	1.70
WO150	162.21	117.03	89.23	1.31	1.81
WO160	162.36	125.98	96.19	1.30	1.68
WO170	161.13	126.4	94.93	1.33	1.69
WP160	161.89	120.06	90.59	1.32	1.78
WP150	162.30	117.17	95.14	1.23	1.70
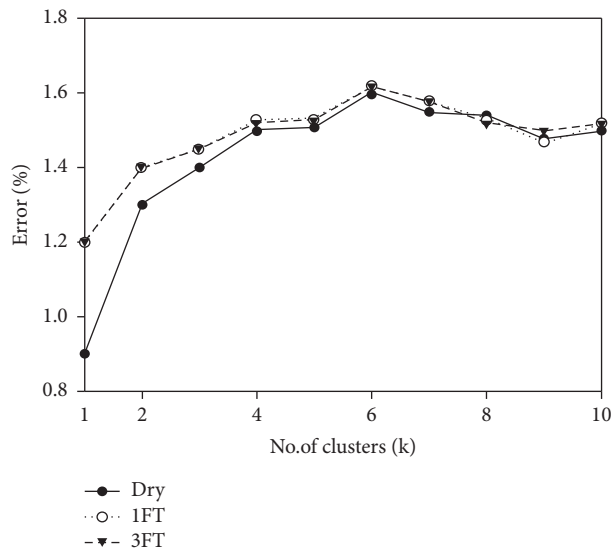
FIGURE 16: Accuracy of PKIP using map-reduce.

However, in terms of single-node implementation, OpenMP may possess lesser degree of fluctuation if compared to Hadoop subjected to the system specification.

4.3. Precision, Recall, and F-Measure Evaluation.

Precision, recall, and F measure are known as the influencing parameters for the evaluation purpose of the proposed PKIP algorithm. The dataset contains 600 image samples comprising of HMA and WMA. The samples were categorized as 300 positive samples and 300 negative samples. The positive samples were the one with adhesion failure, whereas the negative samples were the HMA samples with less than 0.5% adhesion failure. The sample images were cropped into 500×500 pixels and its labelling was done using MS Paint. For the training purpose, the sample images were randomly divided.

Precision is the proportion of genuine positives to the cases that are anticipated as positive. For the proposed experiment, the genuine positives are the pixels that are correctly identified. It is the level of chosen cases that are correct and is computed as follows:

$$\text{Precision} = \frac{\text{correctly identified pixes}}{\text{all detected pixels}}. \quad (14)$$

Recall is the proportion of genuine positives to the all ground truth pixels. The ground truth pixels were the all segmented pixels. Recall is computed as follows:

$$\text{Recall} = \frac{\text{correctly identified pixes}}{\text{all ground truth pixels}}. \quad (15)$$

To determine the weighted average of precision and recall, the parameter of F-measure is used which is computed as follows:

$$F - \text{measure} = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}. \quad (16)$$

For the initial tests, different patch sizes considered as the container of pixels were used with the dimensions 3×3, 5×5, and 9×9. Table 14 shows the impact of patch size on the influencing parameters. It could be clearly observed that the precision, recall, and F value is almost same for 3×3 and 5×5. For 9×9, the precision values are slightly greater than the other two; however, the execution time is substantially high if compared to 3×3 and 5×5. Therefore, for the training evaluation, we have used 3×3 patch for different values of k. The analysis of influencing parameters has been implemented over single node using sequential processing.

Table 15 shows the impact of k value on the influencing parameter, and it could be clearly observed that $k = 10$ gives the highest F value.

4.4. Analysis of Adhesion Failure Using HSV Thresholding.

Tables 16, 17, and 18 represent the percentage of adhesion failure after moisture conditioning (freeze-thaw cycles) and indirect tensile tests for all the k means clustering-based asphalt mixture image samples (i.e., WMA and HMA

prepared with PMD and OPC). The images processed using HSV image thresholding for a WMA sample image dataset and HMA sample image dataset for all three moisture sensitivity test is shown in Figures 17 and 18, where the number of clusters has been set to ten ($k = 10$), and the composition of nonzero and zero pixels were computed using equation (13) to determine the percentage of adhesion failure for all asphalt mixture datasets. For the proposed experiment, Scala version 2.11 programming language has been used for the zero and non-zero-pixel computation.

From Tables 16–18 it becomes evident that the percentage of adhesion failure in HMA and WMA sample subjected to different level of moisture conditioning process depends on mixing temperature and also the type of fillers used. The effect of lower mixing temperature on percent adhesion failure of WMA mixtures has consistent increment in adhesion failure at different levels of moisture conditioning. Moreover, for both the categories, i.e., HMA and WMA, the PMD filler comes out to be better if compared with OPC filler, and in this regard, a difference in adhesion failure for up to 4-5% could be observed in 3FT condition. Section 5 discusses some of the key elements regarding the outcome of the results pertaining to PKIP implementation.

5. Discussion

With respect to the analysis of execution timing of PKIP using C + OpenMP and sequentially, it was inferred from Tables 6–8 that the best execution timings are obtained from the processing of WP150, WP160, and WP170 samples. Further research is required to find the reasoning behind the faster execution time for these samples. From one of the recent studies [2], it is essential to note that in terms of stripping analysis using k means algorithm, the larger execution time arises due to classification of dark shaded areas in the image sample. This reason prevails for the proposed experiment also. Moreover, in order to reduce the execution time of the k means clustering, it is not recommended to downsize the resolution of image sample as it results in loss of pixels which will ultimately affect the accuracy of the final results significantly. From the speedup Table 9, it could also be inferred that a maximum speedup of 1.41X is obtained for WP150 and WP170 for the case of 1FT dataset which in turn implies that execution time using C + OpenMP got reduced up to 30% for these datasets.

After analysing the Map-Reduce implementation of PKIP, it could be inferred from speedup Table 13, that for single node implementation, C + OpenMP-based k means implementation is a viable option if k is lesser than or equal to 5 subjected to the specs of the system. However, if we want to increase the number of clusters up to 10+, then Map-Reduce-based implementation of PKIP algorithm is highly suitable as it can give speedup up to 1.86X, i.e., 46.5% reduction in execution timing for the cases of WP150 and WP170 in 1FT dataset category if compared directly with sequential implementation of k means using C Program. Higher gain in speedup is possible if more compute nodes are added to the cluster. For the proposed experiment, only 3

TABLE 14: Impact of patch size on influencing parameters.

Size of patch	Precision	Recall	F value	Execution time (min)
3 × 3	0.78	0.84	0.80	34
5 × 5	0.78	0.85	0.81	46
9 × 9	0.81	0.87	0.83	87

TABLE 15: Impact of k value on influencing parameters.

k value	Precision	Recall	F value
2	0.75	0.80	0.77
4	0.76	0.81	0.78
6	0.77	0.81	0.78
8	0.78	0.84	0.80
10	0.79	0.84	0.81

TABLE 16: Computation for percentage of adhesion failure for dry condition.

Mixture type	Filler	Mixing temperature	Compaction temperature	Sample	Adhesion failure (%)
HMA	PMD	180	170	HP180	2.68
	OPC	180	170	HO180	2.82
WMA	PMD	170	160	WP170	1.51
	OPC	170	160	WO170	1.76
	PMD	160	150	WP160	1.92
	OPC	160	150	WO160	2.10
	PMD	150	140	WP150	2.33
	OPC	150	140	WO150	2.54

TABLE 17: Computation for percentage of adhesion failure for 1FT condition.

Mixture type	Filler	Mixing temperature	Compaction temperature	Sample	Adhesion failure (%)
HMA	PMD	180	170	HP180	4.20
	OPC	180	170	HO180	4.52
WMA	PMD	170	160	WP170	3.00
	OPC	170	160	WO170	3.30
	PMD	160	150	WP160	3.43
	OPC	160	150	WO160	3.77
	PMD	150	140	WP150	3.65
	OPC	150	140	WO150	3.98

TABLE 18: Computation for percentage of adhesion failure for 3FT condition.

Mixture type	Filler	Mixing temperature (°C)	Compaction temperature (°C)	Sample	Adhesion failure (%)
HMA	PMD	180	170	HP180	9.90
	OPC	180	170	HO180	11.00
WMA	PMD	170	160	WP170	5.10
	OPC	170	160	WO170	9.20
	PMD	160	150	WP160	5.50
	OPC	160	150	WO160	9.80
	PMD	150	140	WP150	5.90
	OPC	150	140	WO150	10.10

additional compute nodes were added. It is essential to note that Hadoop framework has a vast set of scheduling configuration. Therefore, users can exploit these configurations to exploit the maximum resource benefits from Hadoop and thus explore the possibilities of better execution timing results compared to default scheduling algorithms. This also highlights the fact that more room of optimization prevails in the Map-Reduce algorithms. It is deemed necessary to understand that the execution timing plays a prominent role in clustering process. The proposed Map-Reduce-based PKIP algorithm is well suited for clustering large datasets compared to sequential k means algorithm.
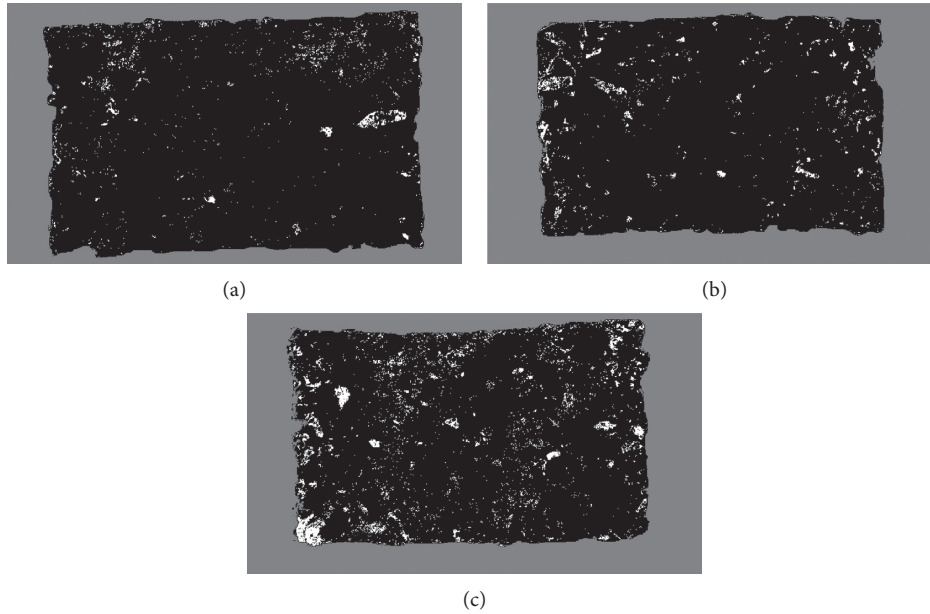
Figure 17: Thresholding applied on clustered samples. (a) Dry-WP170, (b) 1FT-WO170, (c) 3FT-WP170.
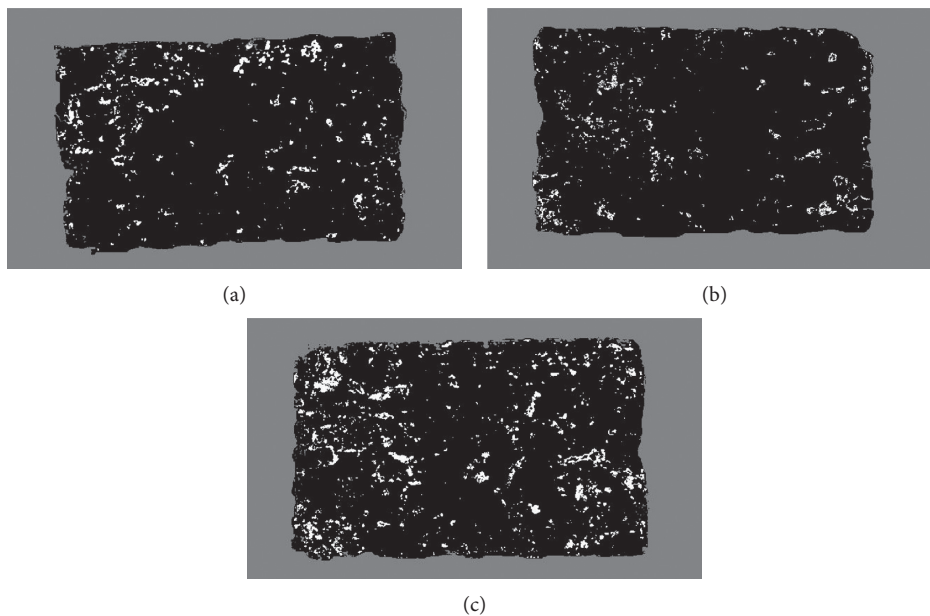
Figure 18: Thresholding applied on clustered samples. (a) Dry-HP180, (b) 1FT-HO180, (c) 3FT-HP180.

It was also be depicted from Tables 16–18 that the WMA specimens prepared using PMD fillers when subjected to 3FT condition show relatively less damage evolution in percentage of adhesion failure if compared to 1FT and dry condition. However, when HMA prepared using PMD filler was subjected to 3FT condition, the evolution of the percentage of adhesion failure was only slightly higher if compared to 1FT and dry conditions. It is to be noted that an environment with indirect light condition should be used in order to prevent light reflections. If such condition is inevitable to avoid, then a heavy usage of advanced filters will be required which will consume higher processing time. A light

blue or cream background is suitable to use as a background for the image analysis, as it becomes easier to differentiate between the foreground and the background prior to the implementation of thresholding process. For the proposed experiment apart from differentiating the stripping using adhesion failure, fractured surfaces due to broken aggregate were also observed. Resultant image sample subjected to different freeze-thaw cycles show that large areas of non-zero pixels (brown color pixels) are reduced with the addition of the Cecabase warm mix additive in general. The results therefore indicate that the surfactant-based chemical additive has worked well to substantially reduce the exposure to

moisture in mixes in particular with the use of PMD filler. In addition, the average stripping rate of WMA blend treated with PMD anti-stripping agents was less than 5.5 percent compared to the maximum level of 11% attained by HMA mixture using OPC filler after 3FT cycles. Amongst the different level of moisture conditioning (dry, 1FT, and 3FT) cycles, both HMA and WMA mixtures with OPC filler have shown the highest percentage of adhesive failure (Figure 1(a)). On the other hand, the use of PMD anti stripping agent in both HMA and WMA has shown relatively lower amount of percent adhesion failure in comparison with OPC filler using different conditioning cycles. The analysis of the results indicates that the amount of coating has been increased in mixes that contain PMD filler using WMA additives irrespective of the conditioning cycles used.

6. Conclusion

From the experimental results, it is concluded that the proposed PKIP algorithm is a viable and effective alternative to assess the percentage of adhesion failure in asphalt mixture and thereby encourages an approach to identify the extent of moisture damage in warm mix asphalt. It is important to be noted that in this experiment, high-resolution (5520×3680) images of asphalt mixtures were used to keep the image quality intact. Therefore, when PKIP algorithm was applied on high-resolution asphalt mixture image using OpenMP and MapReduce, the execution time got reduced up to 30% and 46% if compared with sequential processing of k means clustering. Nonetheless, researchers are encouraged to apply the proposed PKIP algorithm on quadcore and octacore compute nodes to exploit its efficiency to the best possible extent. Moreover, a distributed version of k means clustering can also be applied in 3D image analysis of asphalt mixture by experimenting with different types of filters [39].

From the results obtained in terms of adhesion failure, with respect to HMA, WMA specimens prepared using PMD filler in dry condition have higher moisture resistance. It is also evident that HMA mixtures with OPC filler after three freeze-thaw (3FT) cycles were more prone to moisture damage. However, the WMA mixture with PMD filler was considered more efficient if compared to 1FT and Dry specimens, as the increase in percent adhesion failure was not too high if compared to Dry and 1FT. The percent adhesion failures in WMA mixtures subjected to 3FT cycles were more prevalent than unconditioned mixtures (Dry) and mixtures subjected to 1FT conditioning, confirming that PG-76 based WMA mixtures when subjected to 3FT conditioning compared to HMA were more efficient. Therefore, to analyse the adhesion failure in asphalt mixtures, PKIP algorithm could potentially be used to quantify moisture damage due to its high accuracy and lower execution time.

Appendix

Hue Saturation Value (HSV) Threshold. The *hue* (H) component of any color represents its similarity to any of the purest form of color, while the saturation (S) component of any color shows the non-white element present in it. The value (V) component of any color shows its subsequent darkness or lightness.

k medians is a variation of k means where instead of calculating mean to compute centroid of the cluster, median is computed.

K-medoid minimizes the sum of dissimilarities between points labelled to be in a cluster and a point designated as the center of the cluster.

Note on Hyperthreading. It is essential to note that the CPU having two physical cores have the capability to process four threads parallelly using hyper threading. However, practically, one physical core is designated to run only single thread but with the help of hyper threading, the CPU is capable to exploit the idle pipeline stages to execute another thread. By doing this, it gives an illusion of a separate physical core to the Operating System. Certain sections of the processor are duplicated to store the architectural state, but the main execution resources are not. A basic RISC processor instruction pipeline comprises of five stages, namely, instruction fetch (IF), instruction decode (ID), memory access (MEM), execute (EX), and register write back (WB). One thread can be in the MEM stage and another thread can be in the IF stage. There are some hazards though. If, for example, the MEM stage is dependent on the EX stage, then it could cause problems. Hyper threading aims at minimizing the number of dependent instructions so that another thread may be run in the idle pipeline stages without dependency issues. With respect to OpenMP code parallelization, data points and cluster centres were marked within the shared construct. Moreover, under the shared construct, the iterative matrix multiplication as mentioned in equation (8) third term was implemented using omp for construct.

OpenMP. Library to implement parallel programming for shared-memory-processors.

Otsu's threshold. An algorithm used for image segmentation which returns a single-intensity threshold which separates foreground and background pixels.

$o(n^2 * k * i)$. Here n is the number of objects, k is the number of cluster and i is the number of iterations.

Scala. A *general* purpose programming language which supports both object oriented and functional paradigm of programming.

Data Availability

The input image samples of asphalt mixtures used in this study are available in the Mendeley data repository via the link: https://data.mendeley.com/datasets/x8ptnsjd7x/1 DOI: 10.17632/x8ptnsjd7x.1. The source code for PKIP is available at https://github.com/nishatakhtar/PKIP.

Conflicts of Interest

References

[1] Y. Wu and H. Kim, "Digital imaging in assessment of construction project progress," in *Proceedings of the 21st International Symposium on Automation and Robotics in Construction*, IAARC, Jeju, Korea, pp. 537–542, January 2004.

[2] A. Sahari Moghaddam, E. Rezazadeh Azar, Y. Mejias, and H. Bell, "Estimating stripping of asphalt coating using k-means clustering and machine learning-based classification," *Journal of Computing in Civil Engineering*, vol. 34, no. 1, Article ID 4019044, 2019.

[3] C. Koch, K. Georgieva, V. Kasireddy, B. Akinci, and P. Fieguth, "A review on computer vision based defect detection and condition assessment of concrete and asphalt civil infrastructure," *Advanced Engineering Informatics*, vol. 29, no. 2, pp. 196–210, 2015.

[4] N.-D. Hoang, "Image processing-based recognition of wall defects using machine learning approaches and steerable filters," *Computational Intelligence and Neuroscience*, vol. 2018, Article ID 7913952, 18 pages, 2018.

[5] M. O. Hamzah, S. Y. Teh, B. Golchin, and J. Voskuilen, "Use of imaging technique and direct tensile test to evaluate moisture damage properties of warm mix asphalt using response surface method," *Construction and Building Materials*, vol. 132, pp. 323–334, 2017.

[6] W. Song, B. Huang, and X. Shu, "Influence of warm-mix asphalt technology and rejuvenator on performance of asphalt mixtures containing 50% reclaimed asphalt pavement," *Journal of Cleaner Production*, vol. 192, pp. 191–198, 2018.

[7] M. R. Kakar, M. O. Hamzah, and J. Valentin, "A review on moisture damages of hot and warm mix asphalt and related investigations," *Journal of Cleaner Production*, vol. 99, pp. 39–58, 2015.

[8] S. Xu, F. Xiao, S. Amirkhanian, and D. Singh, "Moisture characteristics of mixtures with warm mix asphalt technologies—a review," *Construction and Building Materials*, vol. 142, pp. 148–161, 2017.

[9] T. Manzur, K. Mahmood Ehsan, S. Lamia Sultana, and S. Mahmud, "Measurement of surface damage through boundary detection: an approach to assess durability of cementitious composites under tannery wastewater," *Advanced*

[10] T. S. Yee and M. O. Hamzah, "Evaluation of moisture susceptibility of asphalt-aggregate constituents subjected to direct tensile test using imaging technique," *Construction and Building Materials*, vol. 227, Article ID 116642, 2019.

[11] M. R. Kakar, M. O. Hamzah, and J. Valentin, "Analyzing the stripping potential of warm mix asphalt using imaging technique," *IOP Conference Series: Materials Science and Engineering*, vol. 236, no. 1, p. 012013, 2017.

[12] M. N. Akhtar, J. M. Saleh, and C. Grelck, "Parallel processing of image segmentation data using Hadoop," *International Journal of Integrated Engineering*, vol. 10, no. 1, 2018.

[13] J. Li, S. Song, Y. Zhang, and Z. Zhou, "Robust k-median and k-means clustering algorithms for incomplete data," *Mathematical Problems in Engineering*, vol. 2016, Article ID 4321928, 8 pages, 2016.

[14] H. Song, J.-G. Lee, and W.-S. Han, "PAMAE: parallel k-medoids clustering with high accuracy and efficiency," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1087–1096, Halifax, Canada, August 2017.

[15] S. Garg, "Variation of k-mean algorithm: a study for high-dimensional large data set," *Information Technology Journal*, vol. 5, pp. 1132–1135, 2006.

[16] M. O. Hamzah, M. R. Kakar, and M. R. Hainin, "An overview of moisture damage in asphalt mixtures," *Jurnal Teknologi*, vol. 73, no. 4, pp. 125–131, 2015.

[17] L. Li, L. Sun, G. Ning, and S. Tan, "Automatic pavement crack recognition based on BP neural network," *Promet-Traffic & Transportation*, vol. 26, no. 1, pp. 11–22, 2014.

[18] A. Cubero-Fernandez, F. J. Rodriguez-Lozano, R. Villatoro, J. Olivares, and J. M. Palomares, "Efficient pavement crack detection and classification," *EURASIP Journal on Image and Video Processing*, vol. 1, p. 39, 2017.

[19] B. Yu, X. Gu, F. Ni, and L. Gao, "Microstructure characterization of cold in-place recycled asphalt mixtures by x-ray computed tomography," *Construction and Building Materials*, vol. 171, pp. 969–976, 2018.

[20] J. Jiang, F. Ni, Q. Dong, Y. Zhao, and K. Xu, "Fatigue damage model of stone matrix asphalt with polymer modified binder based on tensile strain evolution and residual strength degradation using digital image correlation methods," *Measurement*, vol. 123, pp. 30–38, 2018.

[21] N.-D. Hoang, "Detection of surface crack in building structures using image processing technique with an improved otsu method for image thresholding," *Advances in Civil Engineering*, vol. 2018, Article ID 3924120, 10 pages, 2018.

[22] N.-D. Hoang, Q.-L. Nguyen, and D. Tien Bui, "Image processing-based classification of asphalt pavement cracks using support vector machine optimized by artificial bee colony," *Journal of Computing in Civil Engineering*, vol. 32, no. 5, Article ID 4018037, 2018.

[23] J. Schindelin, C. T. Rueden, M. C. Hiner, and K. W. Eliceiri, "The imagej ecosystem: an open platform for biomedical image analysis," *Molecular Reproduction and Development*, vol. 82, no. 7-8, pp. 518–529, 2015.

[24] S. Bhattacharya, S. Gupta, and K. S. Venkatesh, "Dehazing of color image using stochastic enhancement," in *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, pp. 2251–2255, Phoenix, AZ, USA, September 2016.

[25] Y. Chae, M. Nakazawa, and B. Stenger, "Enhancing product images for click-through rate improvement," in *Proceedings of*

the 2018 25th IEEE International Conference on Image Processing (ICIP), pp. 1428–1432, Athens, Greece, October 2018.

[26] S. A. C. Nelson and S. Khorram, *Image Processing and Data Analysis with ERDAS IMAGINE®*, CRC Press, Boca Raton, FL, USA, 2018.

[27] M. O. Hamzah, M. R. Kakar, S. A. Quadri, and J. Valentin, "Quantification of moisture sensitivity of warm mix asphalt using image analysis technique," *Journal of Cleaner Production*, vol. 68, pp. 200–208, 2014.

[28] S. Dong, J. Zhong, P. Hao et al., "Mining multiple association rules in LTPP database: an analysis of asphalt pavement thermal cracking distress," *Construction and Building Materials*, vol. 191, pp. 837–852, 2018.

[29] X. Zheng, Q. Lei, R. Yao, Y. Gong, and Q. Yin, "Image segmentation based on adaptive k-means algorithm," *EURASIP Journal on Image and Video Processing*, vol. 1, p. 68, 2018.

[30] C. Lantieri, R. Lamperti, A. Simone et al., "Use of image analysis for the evaluation of rolling bottle tests results," *International Journal of Pavement Research and Technology*, vol. 10, no. 1, pp. 45–53, 2017.

[31] Z. Li, G. Liu, D. Zhang, and Y. Xu, "Robust single-object image segmentation based on salient transition region," *Pattern Recognition*, vol. 52, pp. 317–331, 2016.

[32] H. Zhu, F. Meng, J. Cai, and S. Lu, "Beyond pixels: a comprehensive survey from bottom-up to semantic image segmentation and cosegmentation," *Journal of Visual Communication and Image Representation*, vol. 34, pp. 12–27, 2016.

[33] T. Liu, C. K. Liyanaarachchi Lekamalage, G.-B. Huang, and Z. Lin, "An adaptive graph learning method based on dual data representations for clustering," *Pattern Recognition*, vol. 77, pp. 126–139, 2018.

[34] M. Al-Ayyoub, S. Al Zubi, Y. Jararweh, M. A. Shehab, and B. B. Gupta, "Accelerating 3D medical volume segmentation using GPUs," *Multimedia Tools and Applications*, vol. 77, no. 4, pp. 4939–4958, 2018.

[35] M. Capó, A. Pérez, and J. A. Lozano, "An efficient approximation to the k-means clustering for massive data," *Knowledge-Based Systems*, vol. 117, pp. 56–69, 2017.

[36] M. Baydoun, H. Ghaziri, and M. Al-Husseini, "CPU and GPU parallelized kernel k-means," *The Journal of Supercomputing*, vol. 74, no. 8, pp. 3975–3998, 2018.

[37] K. Kerdprasop and N. Kerdprasop, "Parallelization of k-means clustering on multi-core processors," in *Proceedings of the 10th WSEAS International Conference on Applied Computer Science*, pp. 472–477, Budapest, Hungary, December 2010.

[38] L. Bai, X. Cheng, J. Liang, H. Shen, and Y. Guo, "Fast density clustering strategies based on the k-means algorithm," *Pattern Recognition*, vol. 71, pp. 375–386, 2017.

[39] M. N. Akhtar, J. M. Saleh, H. Awais, and E. A. Bakar, "Mapreduce based tipping point scheduler for parallel image processing," *Expert Systems with Applications*, vol. 139, Article ID 112848, 2020.

[40] N. Dhanachandra, K. Manglem, and Y. J. Chanu, "Image segmentation using k-means clustering algorithm and subtractive clustering algorithm," *Procedia Computer Science*, vol. 54, pp. 764–771, 2015.

[41] T. Zhang and F. Ma, "Improved rough k-means clustering algorithm based on weighted distance measure with gaussian function," *International Journal of Computer Mathematics*, vol. 94, no. 4, pp. 663–675, 2017.

[42] K. Rajeswari, O. Acharya, M. Sharma, M. Kopnar, and K. Karandikar, "Improvement in k-means clustering algorithm using data clustering," in *Proceedings of the 2015 International Conference on Computing Communication Control and Automation*, pp. 367–369, Pune, India, February 2015.

[43] S. Ding, N. Zhang, J. Zhang, X. Xu, and Z. Shi, "Unsupervised extreme learning machine with representational features," *International Journal of Machine Learning and Cybernetics*, vol. 8, no. 2, pp. 587–595, 2017.

[44] A. Mohan and S. Poobal, "Crack detection using image processing: a critical review and analysis," *Alexandria Engineering Journal*, vol. 57, no. 2, pp. 787–798, 2018.

[45] M. R. Kakar, M. O. Hamzah, M. N. Akhtar, and J. M. Saleh, "Evaluating the surface free energy and moisture sensitivity of warm mix asphalt binders using dynamic contact angle," *Advances in Civil Engineering*, vol. 2019, Article ID 9153603, 15 pages, 2019.

[46] Public Works Department, Standard Specification for Road Works in Malaysia, Section Four: Flexible Pavements, PWD, Malaysia, 2008.

[47] Asphalt Institute, "Superpave mix design," in Superpave Series, No. 2 (SP-2), 3rd ed., Asphalt Institute, Lexington, KY, USA, 2001.

[48] ASTM, ASTM D1559: Test Method for Resistance of Plastic Flow of Bituminous Mixtures Using Marshall Apparatus. ASTM, West Conshohocken, PA, USA, 2006.

[49] ASTM, ASTM D4867: Standard Test Method for Effect of Moisture on Asphalt Concrete Paving Mixtures. ASTM, West Conshohocken, PA, USA, 2006.

[50] D. Chen, N. Roohi Sefidmazgi, and H. Bahia, "Exploring the feasibility of evaluating asphalt pavement surface macrotexture using image-based texture analysis method," *Road Materials and Pavement Design*, vol. 16, no. 2, pp. 405–420, 2015.

[51] R. Chandel and G. Gupta, "Image filtering algorithms and techniques: a review," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 3, no. 10, 2013.

[52] E. Ayguadé, "The design of openMP tasks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 20, no. 3, pp. 404–418, 2008.

[53] T. White, *Hadoop: The Definitive Guide*, O'Reilly Media, Inc., Sebastopol, CL, USA, 2012.

[54] J. Cao, M. Wang, H. Shi, G. Hu, and Y. Tian, "A new approach for large-scale scene image retrieval based on improved parallel -means algorithm in mapreduce environment," *Mathematical Problems in Engineering*, vol. 2016, Article ID 3593975, 17 pages, 2016.

[55] M. Danish, M. N. Akhtar, R. Hashim, J. M. Saleh, and E. A. Bakar, "Analysis using image segmentation for the elemental composition of activated carbon," *MethodsX*, vol. 7, Article ID 100983, 2020.

[56] M. N. Akhtar, J. M. Saleh, and T. Irshad, "Image segmentation using map-reduce framework," in *Proceedings of the 2018 International Conference on Applied Mathematics & Computational Science (ICAMCS. NET)*, Budapest, Hungary, October 2018.

[57] C. Macdonald, R. McCreadie, and I. Ounis, *Agile Information Retrieval Experimentation with Terrier Notebooks*, University of Glasgow, Glasgow, Scotland, 2018.

[58] C. Sreedhar, N. Kasiviswanath, and P. C. Reddy, "Clustering large datasets using k-means modified inter and intra clustering (KM-I2C) in hadoop," *Journal of Big Data*, vol. 4, no. 1, p. 27, 2017.

Research Article

Research on the Prediction of the Water Demand of Construction Engineering Based on the BP Neural Network

Hao Peng (ID),[1] **Han Wu** (ID),[2] **and Junwu Wang** (ID)[2]

[1]*School of Architectural Engineering, Xinyang Vocational and Technical College, Xinyang 464000, China*
[2]*School of Civil Engineering and Architecture, Wuhan University of Technology, Wuhan 430070, China*

Correspondence should be addressed to Han Wu; wuhan20170620@163.com

The scientific and effective prediction of the water consumption of construction engineering is of great significance to the management of construction costs. To address the large water consumption and high uncertainty of water demand in project construction, a prediction model based on the back propagation (BP) neural network improved by particle swarm optimization (PSO) was proposed in the present work. To reduce the complexity of redundant input variables, this model determined the main influencing factors of water demand by grey relational analysis. The BP neural network optimized by PSO was used to obtain the predicted value of the output interval, which effectively solved the shortcomings of the BP neural network model, including its slow convergence speed and easy to fall into local optimum problems. In addition, the water consumption interval data of the Taiyangchen Project located in Xinyang, Henan Province, China, were simulated. According to the results of the case study, there were four main factors that affected the construction water consumption of the Taiyangchen Project, namely, the intraday amount of pouring concrete, the intraday weather, the number of workers, and the intraday amount of wood used. The predicted data were basically consistent with the actual data, the relative error was less than 5%, and the average error was only 2.66%. However, the errors of the BP neural network model, the BP neural network improved by genetic algorithm, and the pluralistic return were larger. Three conventional error analysis tools in machine learning (the coefficient of determination, the root mean squared error, and the mean absolute error) also highlight the feasibility and advancement of the proposed method.

1. Introduction

The cost management of construction engineering is a complex problem, and different cost management strategies should be carried out according to the diverse characteristics of construction materials [1]. The cost management of construction water has the following characteristics. (1) The unit price of water is low and almost constant. Therefore, the essence of cost analysis related to construction water is to analyze the construction water demand. (2) The total amount of water used for construction is huge, and ensuring the stability of the water supply during construction is, therefore, of great significance to the smooth progress of construction [2]. In addition, although the cost of construction water accounts for a small proportion of the total cost of construction projects, water plays a major role in the construction process [3]. The scientific and effective prediction of construction water consumption can not only be used to calculate the cost of construction water scientifically and effectively but can also ensure the stability of the water supply during construction as much as possible, which is also of great significance to the smooth progress of construction.

However, the prediction of construction water consumption is complex. Almost all construction operations require water. Operations such as on-site concrete construction and formwork construction are characterized by high water consumption. Moreover, construction engineering has complex stages, and the water consumption in different construction stages is very different.

The traditional methods for the prediction of industrial water consumption include parametric statistics and

deterministic models, which have usually been incorporated into models for engineering calculation or physical analysis after quantifying the factors that affect construction water consumption. He and Tao [4] established a coupled grey system and multivariate regression model to predict water consumption in Wuhan, which was marked by a clear concept and simple structure. Via the power function model, the linear function model, the logarithmic function model, and the parabolic function model, Zhang et al. [5] fitted the curves of water consumption in China in 2015, 2020, and 2030. The results showed that the correlation coefficient of the parabolic curve fitting was the largest, the average absolute percentage error was the smallest, and the fitting effect was the best. Buck et al. [6] used a statistical sampling method to predict residential water consumption in California. Nevertheless, these research methods were time-consuming and prone to predict errors because they assumed that the relationship between independent and dependent variables was a simple linear process. However, regarding water consumption in construction engineering, the interaction of various variables constitutes a large and compound system with continuous and nonlinear changes. A simplified model would affect the accuracy of the analysis results, thereby resulting in the prediction accuracy of water consumption in construction engineering being unable to meet the needs of engineering practice, and posing a substantial threat to the smooth progress of engineering projects. In addition, traditional water demand forecasting methods need a large amount of complete statistical data to obtain consistent research results. However, hefty amounts of complete statistical data are difficult to obtain on construction sites.

In recent years, the emergence of artificial intelligence algorithms, such as artificial neural networks (ANNs), has provided new ideas for conducting the real-time prediction of the complex system of the water demand of construction projects [7]. The ANN method has a good self-adaptive learning ability and nonlinear mapping ability and is able to fully utilize the potential laws of input data, thereby demonstrating significant advantages for the research and analysis of complex systems with multifactor coupling. When an ANN is applied to the water demand prediction of building engineering construction, the model solves the nonlinear problem of water demand prediction by simulating the structural characteristics and action mechanism of biological neurons and uses the limited data measured in the field instead of a large amount of complete statistical data to predict the water demand by using the data-driven method. At present, relevant scholars have carried out research on water demand forecasting and have obtained rich research results.

Donkor et al. [8] summarized the research results related to urban water demand forecasting and pointed out that scientific and effective forecasting variables are the key to successfully forecasting urban water demand. They also pointed out that the soft computing method yielded valuable research results in the short-term forecasting of water demand. Piasecki et al. [9] compared the ANN and the multiple linear regression (MLR) method, and a case study showed that the ANN method was superior to MLR. Zhang et al. [10] used the main influencing factors of the predicted daily water consumption as the input and the predicted daily water consumption as the output after noise reduction. In this work, the multiscale chaotic genetic algorithm, which is characterized by a strong global searching ability and fast searching speed, was utilized to optimize the parameters of a least-squares support vector machine. By using the ANN method, Santos and Pereira [11] predicted the urban water demand of São Paulo, Brazil. The research indicated that the ANN model could make accurate predictions with a large amount of data, and it was marked by the best performance and a small error. In addition, Santos considered the influences of weather variables on regional urban water consumption. At present, the application of artificial intelligence prediction methods to water consumption prediction mainly includes the following two ideas: a multiparameter prediction model and a time series-based model. Research on the prediction of drinking water demand in Portugal has shown that the univariate time series model based on historical data is useful and can be combined with other prediction methods to reduce errors [12]. The previously mentioned research has demonstrated that soft computing algorithms, such as ANNs, can better deal with the nonlinear problems in water resource demand management. Therefore, the complex and nonlinear mapping between the factors that affect water resource demand and construction water resource demand can be identified by ANNs.

At present, the back propagation (BP) neural network is the most commonly used neural network [13]. When applied to complex system analysis, the traditional BP neural network might contain the following shortcomings. (1) The traditional BP neural network is an optimization method of local search, and it can easily fall into the local extremum. The weights can easily converge to local minima, which leads to the failure of network training [14]. Furthermore, the BP neural network is highly sensitive to initial network weights. When the network is initialized with different weights, it tends to converge to different local minima [15]. (2) The structure of the BP neural network can only be selected by experience, and there is no unified and complete theoretical guidance for the selection of a BP neural network structure. If the selected network structure is too large, the training efficiency will not be high and a fitting phenomenon may occur [16], resulting in low network performance and reduced fault tolerance. If the selection is too small, the network may not converge [17]. Based on the preceding analysis, the traditional BP neural network should be strengthened to develop a high-precision model.

Similar to other metaheuristic algorithms such as the genetic algorithm (GA), particle swarm optimization (PSO) is a population-based optimization tool that searches for the optimal solution by updating generations. PSO has no evolutionary operators such as crossover or mutation. Therefore, the advantages of PSO are its very simple concept, low calculation cost, and few parameters that require adjustment [18]. In the fields of bottom hole pressure prediction, ground vibration prediction [19], and asphaltene

precipitation prediction [20], PSO has been applied to optimize the initial weights and thresholds of the BP neural network, which results in higher accuracy.

In addition, most of the existing research results related to water demand forecasting have been aimed at large-scale areas, such as cities [11–14]. To the best of the authors' knowledge, research on the water demand forecasting for construction projects has not yet been reported.

Hence, in this paper, a prediction model of the water consumption of construction projects was established based on grey relational analysis and the BP neural network improved by PSO. The main contributions of this paper are as follows. (1) Previous relevant research has primarily focused on the water demand prediction of large-scale areas, such as cities. However, the present study focused on the water demand prediction of construction projects in the construction stage and presented a detailed case analysis of the construction water of the Taiyangchen Project in Xinyang City, Henan Province, China. This provided new insight for water management in construction engineering. (2) In this paper, the grey relational analysis method was adopted to identify the key factors that affect the construction water consumption of building engineering and to reduce the input variables of the BP model. By setting the threshold of the grey relational degree, it was determined that the key factors that affected the water consumption in the Taiyangchen Project were the intraday amount of pouring concrete, the intraday weather, the number of workers, and the intraday amount of wood used. (3) In view of the shortcomings of the BP neural network model, such as its slow convergence speed and easy to fall into local optimum, PSO, which is characterized by a fast convergence speed and easy realization, was adopted to optimize the initial weights and thresholds of the BP neural network, which effectively solved these problems. Additionally, the error analysis in the case study demonstrated that the calculation results of the BP neural network improved by PSO achieved higher accuracy than the classical BP neural network model, the BP neural network improved by GA, and the pluralistic return.

The remaining chapters of this paper are arranged as follows. Section 2 presents the materials and methods, and the fundamental factor identification method based on grey relational analysis and the BP neural network optimized by PSO are constructed in detail to build a water demand prediction model for construction projects. Section 3 presents the results and discussion and makes a detailed case analysis of the construction water of the Taiyangchen Project in Xinyang City, Henan Province, China. Three common error analysis tools in machine learning are employed in this section to compare the computational accuracy of different models, and the influence of the topological structure of the BP network model on the calculation results is discussed. Section 4 presents the conclusion, which summarizes the research results of this paper and indicates future research directions.

2. Materials and Methods

2.1. Identification of Key Factors Based on Grey Relational Analysis.
Grey system theory is a systematic scientific theory put forward by Professor Deng Julong, a famous Chinese scholar. Grey relational analysis is a quantitative description and comparison method of system development and changing situations. Its basic concept is tantamount to judge whether factors are closely related by the geometric similarity of reference data columns and several comparison data columns, which reflect the correlation degree between curves. In the fields of risk assessment and prediction, relational analysis can determine the weight of each influencing factor by comparing the compactness of each index series with the benchmark series [21].

The grey correlation method is in a position to analyze the development trend of a system [22]. This method can extract the factors that have great influences on the system index in a system with poor information and small samples. Grey relational analysis can overcome the problems of the calculation amount being too large, the samples not obeying a certain probability distribution, and the calculations having different quantitative and directional results.

The steps for the use of grey relational analysis to find the main factors that affect water consumption in construction projects are reproduced below.

Step 1. Raw data processing.

In this paper, the interval-valued processing method is used to process the original data of construction water and its influencing factors [23].

Step 2. Calculate the grey correlation coefficient.

The degree of correlation can reflect the shape of the sequence, and the coefficient of the grey correlation of water used for building engineering construction is

$$\theta_{mn} = \frac{\rho \Delta_{n\max} + \Delta_{n\min}}{\rho \Delta_{n\max} + \Delta_{mn}(i)}, \tag{1}$$

where $\Delta_{n\max}$ and $\Delta_{n\min}$ are the maximum and minimum values in the water consumption data series of construction projects, respectively, and ρ is a resolution function, the function of which is to improve the significance of the difference between correlation coefficients. Generally, a satisfactory resolution result can be obtained when the value is 0.5 [24]. Moreover, m is the reference sequence, n is the comparison sequence, s is the sequence length, and $\Delta_{mn}(i)$ is the absolute difference between the reference sequence m and point i of the comparison sequence n.

Step 3. Calculate the correlation degree.

The correlation degree calculation formula is as follows [25]:

$$\lambda(x_m, y_n) = \left(\frac{1}{s}\right) \sum_{i=1}^{s} \theta_{mn}(i), \tag{2}$$

where s is the length of the reference sequence, $\theta_{mn}(i)$ is the correlation coefficient between the reference sequence m and the i value of the comparison sequence n, and $\lambda(x_m, y_n)$ is the correlation degree between the reference sequence m on the x curve and the comparison sequence n on the y curve.

Step 4. Rank analysis of correlation degree.

The correlation degree of each factor is sorted depending on the numerical value and describes the relative changes of the reference sequence and comparison sequence. Generally, if the correlation degree between two factors is large, the changes of construction water and the influencing factors are essentially the same [25, 26].

2.2. Prediction Model of Water Demand Based on the BP Neural Network Optimized by PSO

2.2.1. Introduction of the BP Neural Network.
The BP neural network is an artificial neural network model with self-learning and self-adapting abilities and composes of two parts, namely, the forward propagation of input data and the backward propagation of the error value. The standard neural network topology consists of input layer nodes, hidden layer nodes, and output layer nodes that are connected, and the nodes in the same layer do not interact with each other. In this algorithm, n samples $\mathbf{X} = (x_1, x_2, \ldots, x_n)$ are taken as the input nodes of the neural network, and the expected result $\mathbf{Y} = (y_1, y_2, \ldots, y_n)$ is taken as the corresponding output node. The error value can be obtained by comparing the predicted result with the actual result, and the fitness function is employed to measure whether the error value is consistent. For the calculation results that do not meet the requirements, the network will use the gradient descent method to carry out error back propagation in the weight vector space, for which the correction amount of each weight of the hidden layer and the output layer [27] is shown in equation (3). The error reaches the expected value through repeated iteration, thus completing the establishment of the BP neural network calculation model.

$$\mathbf{F}(\psi, \omega, \theta, r) = \left(\sum_{t=1}^{N_1} \sum_{s=1}^{M} \left[y_t(s) - \widehat{y}_t(s) \right] \right)^{-(1/2)}, \quad (3)$$

where ψ is the error value in the BP neural network, ω, θ, and r are the input layer, hidden layer, and output layer of the neural network, respectively, N_1 and M are the weight and threshold number of nodes, respectively, $y_t(s)$ is the expected output of the neural network, $\widehat{y}_t(s)$ is the actual output, t is a node that needs to optimize the connection weight, and s is a node that needs to optimize the threshold.

2.2.2. Optimization of the BP Neural Network by PSO.
PSO was first proposed by Eberhart and Kennedy in 1995 [28]. Its basic concept originated from the study of birds' foraging behavior, and PSO was inspired by this biological population behavior to solve the optimization problem. In PSO, each particle represents a solution of the problem and corresponds to a fitness value. Particle velocity determines the distance and direction of particle motion and is dynamically adjusted by the motion of itself and other particles, thus realizing the optimization process of individuals in a solvable space.

In the process of adopting PSO, the error between the capacity output and the expected capacity output obtained by the forward learning of the BP neural network is first initialized by the PSO to determine the individual extremum and group extremum, i.e., to find the weights and thresholds in the BP neural network. The speed and position are then updated, as are the original individual extremum and group extremum after calculating the fitness. Finally, the obtained optimal neural network weights and thresholds are sent to the BP neural network for verification [29].

Supposing that the particle swarm $\mathbf{X} = (X_1, X_2, \ldots, X_n)$ is composed of n particles, and the dimension of the particles is usually Q. There are n particles in the swarm, each particle is Q-dimensional, and the swarm composed of n particles searches Q dimensions. Every particle is expressed as $X_i = (X_{i1}, X_{i2}, \ldots, X_{iQ})$, which represents the position of the particle i in the Q-dimensional search space and also a potential solution to the problem. According to the objective function, the fitness value corresponding to each particle position X_i can be calculated [30]. The velocity corresponding to each particle can be expressed as $\mathbf{V} = (V_{i1}, V_{i2}, \ldots, V_{iQ})$, and each particle should consider two factors when searching:

(1) The historical optimal value P_i, $\mathbf{P}_i = (P_{i1}, P_{i2}, \ldots, P_{iQ})$, $i = 1, 2, \ldots, n$.

(2) The optimal value P_g, $\mathbf{P}_g = (P_{g1}, P_{g2}, \ldots, P_{gQ})$, found by all particles. It is worth noting that there is only one P_g here.

During each iteration, particles update their own velocity and position via the individual extremum and global extremum, and the update formula of position velocity optimized by the PSO is as follows [31]:

$$v_{id}^{k+1} = \omega v_{id}^{k} + c_1 r \left(p_{id}^{k} - x_{id}^{k} \right) + c_2 r \left(p_{gd}^{k} - x_{id}^{k} \right),$$
$$x_{id}^{k+1} = x_{id}^{k} + v_{id}^{k+1}, \quad (4)$$

where ω is the inertia weight, $d = 1, 2, \ldots, D$, $i = 1, 2, \ldots, n$, k is the current iteration number, v_{id} is the velocity of particles, c_1 is the particle weight coefficient that tracks its own historical optimum value, which represents the particle's own cognition and is called the acceleration factor, c_2 is the weight coefficient of the optimal value of the particle tracking group, which represents the cognition of particles to the whole group knowledge and is called the acceleration factor, and r is a random number that is uniformly distributed in the interval $[0, 1]$. In addition, error analysis should be carried out on the results.

Based on the preceding analysis, the calculation flow chart of the proposed model is presented in Figure 1.

It can be observed in Figure 1 that, via the analysis of the data, the historical data and several factors that have the greatest influences on the water demand of building engineering construction are input into the neural network. After the neurons in each layer act on the influencing factors, they generate the output. The weights and thresholds of the neural network are optimized by PSO, and the fitness value is obtained to determine the individual with the best fitness. After taking the output error as the objective function and
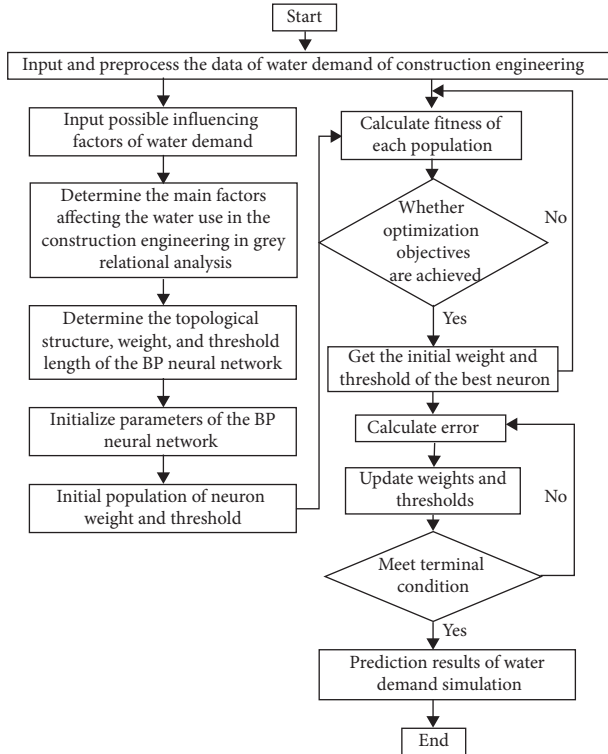
Figure 1: Flow chart of the prediction model of water demand.

correcting the error to meet the requirements, the trained neural network can make predictions.

3. Results and Discussion

3.1. Selection of Influencing Factors. There are many factors that affect water consumption in construction areas [32], such as the number of workers (R_1) [33], the intraday amount of pouring concrete (R_2) [34], the highest temperature (R_3) [35], the intraday weather (R_4) [35], the intraday amount of wood used (R_5) [34], and the intraday amount of steel used (R_6) [34].

In this paper, the data of the Taiyangchen Project in May and June 2019 were collected in units of days. The data of the water use of the Taiyangchen Project was processed by grey relational analysis, and the correlation coefficient and degree were obtained. By comparing the sizes, the main factors that affected the water use of the Taiyangchen Project were determined and then used as input layers and input into the neural network to predict the water use of the project.

The daily water consumption of the Taiyangchen Project in Xinyang City, Henan Province, China, is the research object in this work. The project consists of nine residential buildings with frame-shear wall structures, all of which have 18 floors above ground and 1 floor underground. The height of each building is 52.90 m, and the total construction area of each building is 13449.68 m^2.

The quantification of the evaluation factors is an important step in the selection and treatment of influencing factors. The number of workers (R_1), the intraday amount of pouring concrete (R_2), the highest temperature (R_3), the

intraday amount of wood used (R_5), and the intraday amount of steel used (R_6) were all determined by actual field investigation and statistics. The score of the intraday weather (R_4) is divided into four situations, namely, sunny (0.9), cloudy (0.6), light rain (0.3), and heavy rain (0). The values of each factor are presented in Table 1. Due to the limitations of the layout, only some samples are reported in Table 1.

The correlation coefficients of the influencing factors were calculated by equations (1) and (2), and the calculation results are shown in Table 2.

It can be seen from Table 2 that the correlation degree of the influencing factors from the greatest to the least is as follows: the intraday amount of pouring concrete (R_2) > the intraday weather (R_4) > the number of workers (R_1) > the intraday amount of wood used (R_5) > the highest temperature (R_3) > the intraday amount of steel used (R_6). This order can be explained by the content and characteristics of the construction work. Concrete pouring is a typical wet operation that requires a lot of water. The weather is another significant factor that affects construction. When it rains, most of the work on the construction site will stop, and the construction water consumption will decrease significantly. The more workers there are, the more water will be used for construction and living. Timber for construction needs to be watered and wetted to ensure that its moisture content is near the optimum, which also requires a large amount of water [36].

When the correlation degree is less than 0.6, the two sequences are considered to be independent, and if the correlation degree is greater than 0.8, the two sequences have a good correlation. A correlation value between 0.6 and 0.8 is beneficial [35]. In Table 2, the factors with a correlation degree greater than 0.8 include the intraday amount of pouring concrete (R_2), the intraday weather (R_4), the number of workers (R_1), and the intraday amount of wood used (R_5), and these are, therefore, the key factors that affect the water consumption of building engineering construction.

3.2. Result of Water Demand Forecasting. In this work, the construction water consumption of the Taiyangchen Project in Xinyang City, Henan Province, China, was taken as the research object, and the data used were sourced from the monitoring data of the municipal pipe network water consumption of the Taiyangchen Project and the on-site construction log.

After the evaluation factor is quantified, the difference of its numerical dimension slows the convergence speed of the algorithm and affects the accuracy of the model. Because all indicators are benefit-based, the data are normalized as follows [37]:

$$x_{ij}^* = \frac{x_{ij} - \min(x_j)}{\max(x_j) - \min(x_j)}, \quad (5)$$

where x_{ij}^* indicates the value of the evaluation index after standardization, $\max(x_j)$ represents the maximum value of indicator j, and $\max(x_j)$ represents the minimum value of indicator j.

TABLE 1: Numerical value of each factor and actual water consumption.

No.	R_1	R_2	R_3	R_4	R_5	R_6	Actual water consumption
Unit	—	m^3	°C	—	m^3	t	t
1 May	228	310.00	28	0.3	13.00	73.46	836.171
2 May	198	1050.00	24	0.6	57.10	72.67	1200.363
3 May	219	1200.00	26	0.6	9.10	83.07	1428.350
4 May	255	730.00	28	0.6	3.10	165.73	567.857
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
27 June	229	50.00	36	0.3	36.10	84.41	752.873
28 June	200	150.00	35	0.3	3.50	44.68	758.394
29 June	215	780.00	32	0.6	18.30	36.38	1601.371
30 June	219	250.00	32	0.3	4.90	107.60	702.361

TABLE 2: Correlation coefficients of influencing factors.

Factor	R_1	R_2	R_3	R_4	R_5	R_6
Correlation coefficient	0.8454	0.8927	0.5150	0.8589	0.8117	0.4625
Mark	Reserved	Reserved	Deleted	Reserved	Reserved	Deleted

According to grey relational analysis, four main influencing factors of water demand in the construction interval of building engineering were obtained. The number of input nodes is $m = 4$, and the number of hidden layer nodes is $n = 2m + 1 = 9$, so the structure of the BP neural network is 4-9-1 [38], as illustrated in Figure 2. It is worth noting that a variety of BP neural network topologies were constructed in this work, and the calculation results are exhibited in Table 3.

In the process of model formation using the BP neural network, the available data should be divided into two groups, which, respectively, represent training and testing sets. The data of the training set are used for training, while the data of the testing set are used for checking the network. Many researchers choose data in the respective proportions of 90% and 10%, 80% and 20%, or 70% and 30% [39]. In this study, the training set was data of the Taiyangchen Project from May 1 to June 20, 2018, including a total of 51 days of data. The testing set was the data of the Taiyangchen Project from June 21 to June 30, 2018, including a total of 10 days of data. The ratio of training set data to testing set data was therefore 83.61%–16.39%.

To achieve a better prediction effect, the best parameters of the BP neural network and the PSO were set, including the following: the number of training iterations was 1000, the learning rate was 0.1, and the training target was 0.001. The calculation parameters [40] of PSO included 1000 iterations, a population size of 50, the local learning factor $c_1 = 1.49445$, and the global learning factor $c_2 = 1.49445$. The maximum error of iteration termination was 0.00001.

The convergence curve obtained after calculation is presented in Figure 3. In a case analysis, when the number of iterations reaches about 500, the requirements are met. In this study, the population number and the maximum number of iterations were set to relatively large values to ensure that the model could calculate more complex problems. Figure 3 shows the convergence curve after 1000 iterations.

Following the optimization calculation process of PSO, the error between the 498th iteration and 499th iteration was
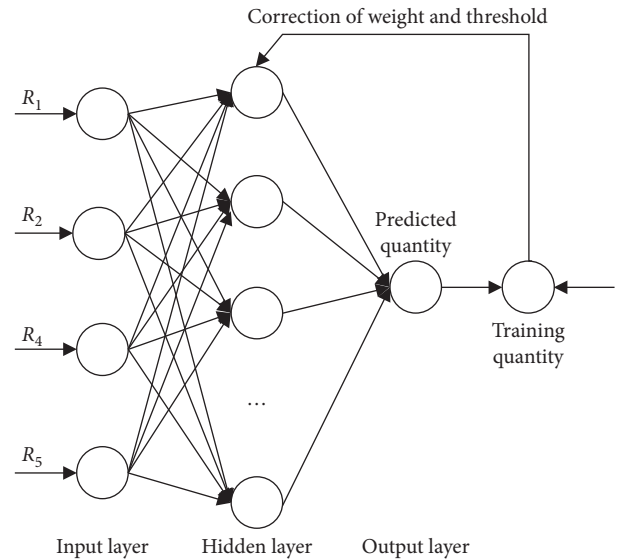
FIGURE 2: Topological structure diagram of the BP neural network.

TABLE 3: Comparison of three error representations.

Error representations	R^2	RMSE	MAE
BP	0.7921	731.2692	45.4554
PSO-BP	0.9959	96.0900	15.7467
GA-BP	0.9853	130.1673	20.3815
Pluralistic return	0.3767	1938.1279	73.0130

greater than the minimum acceptable precision (0.00001), while the error between the 499th iteration and 500th iteration was less than 0.00001. After that, the errors of the calculation results were all less than 0.00001. The calculation was arrested at the 1000th iteration with a very small error. These findings indicate that the PSO found the optimal neural network weights and thresholds at the 500th iteration, which is illustrated by both Figure 3 and Table 4. In addition, it can be qualitatively judged from Figure 3 that the GA
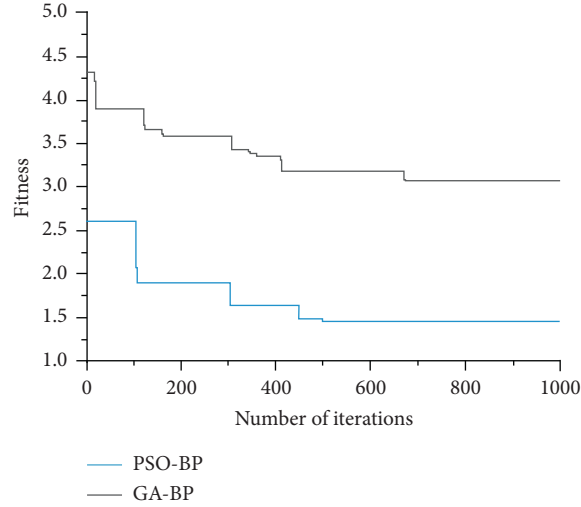
FIGURE 3: The convergence curve of PSO and GA.

TABLE 4: Precision level and calculation termination of PSO at the 1000th iteration.

Iteration (n)	Fitness ($n-1$)	Fitness (n)	Fitness (n) – Fitness ($n-1$)	Result
498	1.488344	1.488344	$0 < 0.00001$	Continue
499	1.488344	1.455848356	$0.032495395 > 0.0001$	Continue
500	1.455848356	1.455848356	$0 < 0.0001$	Continue
1000	1.455848	1.455848	$0 < 0.0001$	Stop

converged between 700 and 800 generations. The PSO converged faster than GA. This is an advantage of PSO compared with GA.

The prediction results obtained by using the proposed calculation model are reported in Table 5. In addition, the BP model, the pluralistic return, and the BP model improved by GA were also used to predict the results. The parameters of GA were set as follows: population size was 100, ending evolution algebra was 1000, crossover probability was 0.5, and mutation probability was 0.001.

In addition, this paper used the standard calculation method which was commonly used in engineering practice, and multivariate linear return method to calculate the quantity used in the construction site.

According to Chinese national standard (the Code for Fire Protection Design of Buildings, GB 50016-2014), the calculation process of construction site water consumption is as follows:

(1) Water consumption for site operation:

$$q_1 = K_1 \sum \left(\frac{Q_1 \cdot N_1}{T_1 \cdot t} \right) \cdot \left(\frac{K_2}{8 \times 3600} \right), \tag{6}$$

where q_1 (L/s) is the water consumption of construction site, K_1 is the unexpected construction water consumption coefficient, Q_1 (L/s) is the annual engineering quantity, N_1 is the construction water quota (L/m^3), T_1 (days) is the annual effective working day, t (hours) is the number of working shifts per day, and K_2 is the imbalance coefficient of water consumption.

According to the field investigation, K_1 was 1.10, T_1 was 300, and t was 8 in this project. The calculated Q_1 and N_1 were brought into equation (6), and the q_1 was 0.52 L/s.

(2) Water for construction machinery:

$$q_2 = K_1 \sum Q_2 \cdot N_2 \cdot \left(\frac{K_3}{8 \times 3600} \right), \tag{7}$$

where q_2 is the water consumption of machinery, K_1 is the unexpected construction water consumption coefficient, Q_2 is the same number of machinery, N_2 is the water quota of construction machinery machine-team, and K_3 is the water imbalance coefficient of construction machinery.

According to field investigation, K_3 was 1.5. According to the construction site statistics, Q_2 and N_2 were taken into equation (8), and $q_2 = (0.04$ L/s) could be calculated.

(3) Domestic water consumption on the construction site:

$$q_3 = \frac{P_1 \cdot N_3 \cdot K_4}{t \times 8 \times 3600}, \tag{8}$$

where q_3 is the domestic water consumption of the construction site, P_1 is the domestic water

TABLE 5: Prediction results of different models.

Time	Actual water consumption	BP	PSO-BP	GA-BP	Pluralistic return
21 June	1430.37	1397.56	1446.50	1449.50	967.29
22 June	860.32	605.91	857.48	883.10	554.43
23 June	610.36	702.29	581.46	633.88	571.73
24 June	604.13	876.12	615.22	557.52	516.61
25 June	1560.54	1367.86	1528.54	1619.99	463.06
26 June	1910.00	2315.52	1888.37	1876.97	468.56
27 June	752.87	618.75	786.22	779.86	485.53
28 June	758.39	722.45	763.90	746.23	420.71
29 June	1601.37	1409.33	1668.54	1719.87	551.14
30 June	702.36	1029.35	673.01	655.37	450.15

consumption of the construction site, N_3 is the water quota of the construction site, K_4 is the imbalance coefficient of the construction site, and t is the number of working shifts per day.

According to the field investigation, K_4 was 1.3 and t was 1. With equation (8), q_3 was 2.08 L/s.

(4) Domestic water consumption in living quarters:

$$q_4 = \frac{P_2 \cdot N_4 \cdot K_5}{24 \times 3600}, \tag{9}$$

where q_4 is the domestic water consumption in the living area, P_2 is the number of residents in the living area, N_4 is the daily domestic water quota in the living area, and K_5 is the unbalanced coefficient of water consumption in the living area.

After field calculation, P_2 was 150 and K_5 was 2. Bringing complex N_4 into equation (9), q_4 was 0.8 L/s.

(5) Total water consumption Q:

$$Q = q_1 + q_2 + q_3 + q_4. \tag{10}$$

Bring q_1, q_2, q_3, and q_4 into equation (10), and the Q is 3.44 L/s. It is worth mentioning that the fire water was not considered in our calculation here because the fire water was only used in the fire.

Considering that the daily construction time is 8 hours, the water consumption calculated according to Chinese construction codes was 990.72 m^3. Comparing the data in Tables 1 and 5, the water consumption calculated by Chinese construction codes was often less than the actual water demand. This situation would easily lead to water shortage and shutdown in the construction site, which was one of the important backgrounds of the research work in this paper. In addition, under the constraints of limited time and timeliness of data collection, it took a long time to investigate the values of more than a dozen parameters by adopting Chinese national calculation standard, which had the disadvantage of low calculation efficiency.

Using the return analysis function of Excel 2016 Software, the expression of multivariate return was calculated as follows:

$$Q = 211.1739 + 0.8247r_1 + 0.1152r_2 + 72.2757r_4 + 1.6083r_5. \tag{11}$$

Bring the data of the test set into equation (11), and the prediction result is as shown in Table 5.

3.3. Analysis and Discussion of Calculation Results. Error analysis was conducted to verify the accuracy of this algorithm, and the relative error value was obtained according to the predicted and actual values. The formula is as follows:

$$E = \frac{\left|\gamma_p - \gamma_a\right|}{\gamma_a}, \tag{12}$$

where E is the relative error, γ_p is the predicted value, and γ_a is the true value.

According to equation (12), the relative error value can be calculated by the values present in Table 6. Compared with the actual water consumption, the error of the calculations of the proposed method was less than 5%, and the average error was only 2.47%. The average error of the GA-BP model was 4.06%, and the maximum error was 8.36%. In contrast, the error of the calculations of the BP model was larger, the maximum error was 46.56%, and the average error was 22.39%. The maximum error of the results calculated by multivariate return analysis was 75.47%, and the average error was 41.61%. This proves that the proposed method is effective and advanced in predicting the water demand of construction projects.

The four calculation results of PSO-BP with the biggest error appeared in the last four times. That was to say, the first six predictions were very accurate, and the last four predictions had large errors.

To better compare the prediction results of the two methods and highlight the advantages of the proposed PSO-BP neural network model, three common error analysis tools in machine learning were used to compare several algorithms.

TABLE 6: Prediction results of different models.

Time	BP (%)	PSO-BP (%)	GA-BP (%)	Pluralistic return (%)
21 June	2.29	1.13	1.32	32.38
22 June	26.10	0.33	2.58	35.55
23 June	15.06	0.73	3.71	6.33
24 June	45.02	1.84	8.36	14.49
25 June	12.35	2.05	3.67	70.33
26 June	21.23	1.13	1.76	75.47
27 June	17.82	4.43	3.46	35.51
28 June	25.53	4.74	1.63	44.53
29 June	11.99	4.19	6.89	65.58
30 June	46.56	4.18	7.17	35.91

TABLE 7: Calculation results of topological structures of different network models.

Model	Number of hidden layers	Number of hidden layer nodes	Average error
BP	1	9	11.66
	2	9-5	17.37
	2	9-7	9.53
GA-BP	1	9	4.06
	2	9-5	3.65
	2	9-7	2.37
PSO-BP	1	9	2.47
	2	9-5	1.38
	2	9-7	2.31

TABLE 8: Calculation results with different numbers of input variables.

Input variables	Average error (%)	Maximum error (%)
r_1, r_2, r_4, and r_5	4.06	4.74
r_1, r_2, and r_4	7.43	10.15
r_1, r_2, and r_5	9.55	14.28
r_1, r_4, and r_5	8.90	15.69
r_2, r_4, and r_5	7.05	12.73
r_1, r_2, r_3, r_4, and r_5	3.67	4.21
r_1, r_2, r_3, r_4, r_5, and r_6	3.31	3.96

The determination coefficient (R^2) indicates the degree of correlation between the measured value and the predicted value. The closer R^2 is to 1, the higher the correlation. On the contrary, the closer R^2 is to 0, the lower the correlation. As listed in Table 3, the R^2 values of the BP model, the PSO-BP model, the GA-PSO model, and the pluralistic return were found to be 0.7921, 0.9959, 0.9853, and 0.3767, respectively, thereby demonstrating that the PSO-BP model is better than the BP model, the GA-BP model, and the pluralistic return. The root mean square error (RMSE) is an important standard by which to measure the prediction results of machine learning models. The RMSE of the PSO-BP model was 96.0900, which is notably less than the BP model, the GA-BP model, and the pluralistic return. The mean absolute error (MAE) is the average of absolute errors, which can better reflect the actual situation of predicted value errors. The RMSE of the PSO-BP model was 15.7467, which is notably less than the BP model, GA-PSO, and pluralistic return. The comparative analysis of calculation errors indicates that the PSO-BP model achieved better prediction accuracy and optimization performance. This excellent calculation result is also consistent with previous benchmark test results [41].

The number of hidden layers is another important factor that affects the accuracy of the BP neural network [42]. Therefore, the influences of the topological structures of different network models on the prediction results were also analyzed [38].

Referring to the classical research results in related fields [43], the topological structures of six different network models were designed, and the related calculation results are shown in Table 7.

It can be seen from Table 3 that the calculation accuracy of the PSO-BP neural network model was significantly higher than that of the BP model or GA-BP model when the same network model topology was adopted. Regardless of the topology of the network model, the average error of the calculation results of the PSO-BP neural network model was very small. The calculation results demonstrate the effectiveness and advancement of the PSO-BP neural network model in forecasting the water demand of construction projects. In addition, overfitting or underfitting is a qualitative phenomenon that occurs in artificial neural network algorithms, and there is no tool with which to quantitatively describe them. Referring to the details of previous research studies [41, 44], it is believed that the prediction accuracy of the PSO-

BP neural network model is higher than the calculation accuracy of the BP neural network model, which also demonstrates that the proposed model reduces the phenomenon of overfitting or underfitting. After overfitting or underfitting, the prediction accuracy is often inadequate [45].

Regarding the PSO-BP neural network model, with the increase of the number of hidden layers, the average error was considered to be further reduced. There are two hidden layers, and the model with the 4-9-5-1 network structure achieved the highest calculation accuracy. The accuracy of calculation with a hidden layer model was 2.47%, which was the lowest in the PSO-BP neural network model and can also meet the needs of engineering practice [46].

This paper discussed the influence of the number of input variables on the calculation results. The analysis here adopted the calculation model of PSO-BP, and the number of hidden layer was 1. The average error and maximum error are shown in Table 8.

When there were three input variables, the calculation error of the PSO-BP model was obviously larger than that of four input variables. When the input variables were increased to 5 or 6, the calculation accuracy was not significantly improved. Considering the availability of construction site data, the field investigation time would increase significantly if there were 5 or 6 input variables. Therefore, it could be considered that the four input variables obtained by grey relational analysis in this paper were reasonable.

4. Conclusions

The purpose of this research was to use the BP neural network to accurately predict the water consumption of

construction projects. First, via the use of the data, it was found that there are four factors that affect the water consumption in construction projects, namely, the intraday amount of pouring concrete, the intraday weather, the number of workers, and the intraday amount of wood used. Then, after taking these four key factors as the input layer and using the optimal neural network weights and thresholds obtained by PSO, a predictive model of construction water consumption based on the neural network model was constructed. Finally, a case study of the construction water consumption of the Taiyangchen Project in Xinyang City, Henan Province, China, revealed that, compared with the actual water consumption, the error calculated by the proposed method was less than 5%, and the average error was only 2.47%. In addition, the three common error analysis tools used in machine learning (the coefficient of determination, the root mean squared error, and the mean absolute error) all highlighted that the calculation accuracy of the proposed method was significantly higher than the BP algorithm, the GA-BP, and the pluralistic return. There were two hidden layers in the PSO-BP neural network model, and the model with the 4-9-5-1 network structure was found to have the highest calculation accuracy. The calculation accuracy of the model with a hidden layer and a network structure of 4-9-1 was 2.47%, which can also meet the needs of engineering practice. The model proposed in this paper can effectively predict the water consumption of building engineering construction and determine abnormal water in a timely manner to rationally dispatch the water supply and ultimately achieve the purpose of saving water. In future research, the sample set can be expanded, the learning effect of the model can be improved, and a more perfect prediction model of construction water consumption can be trained.

Data Availability

The case analysis data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

References

[1] J.-S. Chou, "Cost simulation in an item-based project involving construction engineering and management," *International Journal of Project Management*, vol. 29, no. 6, pp. 706–717, 2011.

[2] I. Ghalehkhondabi, E. Ardjmand, W. A. Young et al., "Water demand forecasting: review of soft computing methods," *Environmental Monitoring and Assessment*, vol. 189, no. 7, Article ID 313, 2017.

[3] P. W. Jayawickrama and S. Rajagopalan, "Alternative water sources in Earthwork construction," *Transportation Research Record*, vol. 2004, pp. 88–96, 2007.

[4] F. He and T. Tao, "An improved coupling model of grey-system and multivariate linear regression for water consumption forecasting," *Polish Journal of Environmental Studies*, vol. 23, no. 4, pp. 1165–1174, 2014.

[5] X. Zhang, M. Yue, Y. Yao, and H. Li, "Regional annual water consumption forecast model," *Desalination and Water Treatment*, vol. 114, pp. 51–60, 2018.

[6] S. Buck, M. Auffhammer, H. Soldati et al., "Forecasting residential water consumption in California: rethinking model selection," *Water Resources Research*, vol. 56, no. 1, 2020.

[7] X. Wu, V. Kumar, J. Ghosh et al., "Top 10 algorithms in data mining," *Knowledge and Information Systems*, vol. 14, no. 1, pp. 1–37, 2008.

[8] E. A. Donkor, T. A. Mazzuchi, R. Soyer, and J. Alan Roberson, "Urban water demand forecasting: review of methods and models," *Journal of Water Resources Planning and Management*, vol. 140, no. 2, pp. 146–159, 2014.

[9] A. Piasecki, J. Jurasz, and B. Kazmierczak, "Forecasting daily water consumption: a case study in town, Poland," *Periodica Polytechnica-Civil Engineering*, vol. 62, no. 3, pp. 818–824, 2018.

[10] W. Zhang, Q. Yang, M. Kumar, and Y. Mao, "Application of improved least squares support vector machine in the forecast of daily water consumption," *Wireless Personal Communications*, vol. 102, no. 4, pp. 3589–3602, 2018.

[11] C. C. does Santos and A. J. Pereira, "Water demand forecasting model for the metropolitan area of São Paulo, Brazil," *Water Resources Management*, vol. 28, no. 13, pp. 4401–4414, 2014.

[12] A. Sardinha-Lourenço, A. Andrade-Campos, A. Antunes, and M. S. Oliveira, "Increased performance in the short-term water demand forecasting through the use of a parallel adaptive weighting strategy," *Journal of Hydrology*, vol. 558, pp. 392–404, 2018.

[13] I. A. Basheer and M. Hajmeer, "Artificial neural networks: fundamentals, computing, design, and application," *Journal of Microbiological Methods*, vol. 43, no. 1, pp. 3–31, 2000.

[14] Z. Zhao, Q. Xu, and M. Jia, "Improved shuffled frog leaping algorithm-based BP neural network and its application in bearing early fault diagnosis," *Neural Computing and Applications*, vol. 27, no. 2, pp. 375–385, 2016.

[15] Z. Ma, X. Song, R. Wan, L. Gao, and D. Jiang, "Artificial neural network modeling of the water quality in intensive Litopenaeus vannamei shrimp tanks," *Aquaculture*, vol. 433, pp. 307–312, 2014.

[16] P. Bansal, S. Gupta, S. Kumar et al., "MLP-LOA: a metaheuristic approach to design an optimal multilayer perceptron," *Soft Computing*, vol. 23, no. 23, pp. 12331–12345, 2019.

[17] S. Yang, X. Zhu, L. Zhang, L. Wang, and X. Wang, "Classification and prediction of Tibetan medical syndrome based on the improved BP neural network," *IEEE Access*, vol. 8, pp. 31114–31125, 2020.

[18] D. T. Bui, N. D. Hoang, T. D. Pham et al., "A new intelligence approach based on GIS-based multivariate adaptive regression splines and metaheuristic optimization for predicting flash flood susceptible areas at high-frequency tropical typhoon area," *Journal of Hydrology*, vol. 575, pp. 314–326, 2019.

[19] D. J. Armahani, M. Hajihassani, E. T. Mohamad et al., "Blasting-induced flyrock and ground vibration prediction through an expert artificial neural network based on particle swarm optimization," *Arabian Journal of Geosciences*, vol. 7, no. 12, pp. 5383–5396, 2014.

[20] M. A. Ahmadi and S. R. Shadizadeh, "New approach for prediction of asphaltene precipitation due to natural depletion by using evolutionary algorithm concept," *Fuel*, vol. 102, pp. 716–723, 2012.

[21] Z. Zang, Y. Wang, and Z. X. Wang, "A grey TOPSIS method based on weighted relational coefficient," *Journal of Grey System*, vol. 26, no. 2, pp. 112–123, 2014.

[22] Z. X. Wang, "Correlation analysis of sequences with interval grey numbers based on the kernel and greyness degree," *Kybernetes*, vol. 42, no. 1-2, pp. 309–317, 2013.

[23] G. Deng, J. Qiu, G. Liu, and K. Lv, "Environmental stress level evaluation approach based on physical model and interval grey association degree," *Chinese Journal of Aeronautics*, vol. 26, no. 2, pp. 456–462, 2013.

[24] N. M. Xie and S. F. Liu, "Research on evaluations of several grey relational models adapt to grey relational axioms," *Journal of Systems Engineering and Electronics*, vol. 20, no. 2, pp. 304–309, 2009.

[25] B. Zhu, L. Yuan, and S. Ye, "Examining the multi-timescales of European carbon market with grey relational analysis and empirical mode decomposition," *Physica A: Statistical Mechanics and its Applications*, vol. 517, pp. 392–399, 2019.

[26] K. Zhang, Y. Chen, and L. F. Wu, "Grey spectrum analysis of air quality index and housing price in Handan," *Complexity*, vol. 2019, Article ID 8710138, , 2019.

[27] S. M. Huang, "Production capacity prediction based on neural network technology in an efficient economic and management environment of oil field," *Fresenius Environmental Bulletin*, vol. 29, no. 4, pp. 2442–2449, 2020.

[28] C. A. C. Coello, G. T. Pulido, and M. S. Lechuga, "Handling multiple objectives with particle swarm optimization," *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 3, pp. 256–279, 2004.

[29] C. Hou, X. Yu, Y. Cao, C. Lai, and Y. Cao, "Prediction of synchronous closing time of permanent magnetic actuator for vacuum circuit breaker based on PSO-BP," *IEEE Transactions on Dielectrics and Electrical Insulation*, vol. 24, no. 6, pp. 3321–3326, 2017.

[30] Z. H. Zhan, J. Zhang, Y. Li et al., "Adaptive particle swarm optimization," *IEEE Transactions on Systems Man and Cybernetics Part B-Cybernetics*, vol. 39, no. 6, pp. 1362–1381, 2009.

[31] F. vandenBergh and A. P. Engelbrecht, "A cooperative approach to particle swarm optimization," *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 3, pp. 225–239, 2004.

[32] A. Piasecki, J. Jurasz, and W. Marszelewski, "Application of multilayer perceptron artificial neural networks to mid-term water consumption forecasting - a case study," *Ochrona Srodowiska*, vol. 38, no. 2, pp. 17–22, 2016.

[33] M. E. Banihabib and P. Mousavi-Mirkalaei, "Extended linear and non-linear auto-regressive models for forecasting the urban water consumption of a fast-growing city in an arid region," *Sustainable Cities and Society*, vol. 48, Article ID 101585, 2019.

[34] M. Y. Han, G. Q. Chen, J. Meng, X. D. Wu, A. Alsaedi, and B. Ahmad, "Virtual water accounting for a building construction engineering project with nine sub-projects: a case in E-town, Beijing," *Journal of Cleaner Production*, vol. 112, pp. 4691–4700, 2016.

[35] L. N. Yang, W. Z. Li, and X. Liu, "On campus interval water demand prediction based on grey genetic BP neural network," *Journal of Water Resources and Water Engineering*, vol. 30, no. 3, pp. 133–138, 2019.

[36] R. Walker, S. Pavía, and M. Dalton, "Measurement of moisture content in solid brick walls using timber dowel," *Materials and Structures*, vol. 49, no. 7, pp. 2549–2561, 2016.

[37] D. Liu, J. P. Feng, H. Li et al., "Spatiotemporal variation analysis of regional flood disaster resilience capability using an improved projection pursuit model based on the wind-driven optimization algorithm," *Journal of Cleaner Production*, vol. 241, Article ID 118406, 2019.

[38] W. Cheng, M. Zhou, J. J. Ye et al., "PSO-BP modeling research on fee rate measurement of construction project safe construction cost," *China Safety Science Journal*, vol. 26, no. 5, pp. 146–151, 2016.

[39] G. Zhang, B. Eddy Patuwo, and M. Y. Hu, "Forecasting with artificial neural networks," *International Journal of Forecasting*, vol. 14, no. 1, pp. 35–62, 1998.

[40] I. C. Trelea, "The particle swarm optimization algorithm: convergence analysis and parameter selection," *Information Processing Letters*, vol. 85, no. 6, pp. 317–325, 2003.

[41] H. H. Li, Y. D. Lu, C. Zheng et al., "Groundwater level prediction for the arid oasis of Northwest China based on the artificial Bee Colony algorithm and a back-propagation neural network with double hidden layers," *Water*, vol. 11, no. 4, Article ID 860, 2019.

[42] A. Kaveh and H. Servati, "Design of double layer grids using backpropagation neural networks," *Computers & Structures*, vol. 79, no. 17, pp. 1561–1568, 2001.

[43] K. M. Neaupane and S. H. Achet, "Use of back propagation neural network for landslide monitoring: a case study in the higher Himalaya," *Engineering Geology*, vol. 74, no. 3-4, pp. 213–226, 2004.

[44] Q. Shen, W.-m. Shi, X.-p. Yang, and B.-x. Ye, "Particle swarm algorithm trained neural network for QSAR studies of inhibitors of platelet-derived growth factor receptor phosphorylation," *European Journal of Pharmaceutical Sciences*, vol. 28, no. 5, pp. 369–376, 2006.

[45] X. F. Yan, "Hybrid artificial neural network based on BP-PLSR and its application in development of soft sensors," *Chemometrics and Intelligent Laboratory Systems*, vol. 103, no. 2, pp. 152–159, 2010.

[46] F. Y. Jiang, Y. L. Zhao, S. Dong et al., "Prediction of submarine pipeline damage based on genetic algorithm," *Transactions of Oceanology and Limnology*, vol. 3, pp. 52–59, 2019.

Research Article

Practical Hybrid Machine Learning Approach for Estimation of Ultimate Load of Elliptical Concrete-Filled Steel Tubular Columns under Axial Loading

Tien-Thinh Le [1,2]

[1]*Faculty of Mechanical Engineering and Mechatronics, Phenikaa University, Yen Nghia, Ha Dong, Hanoi 12116, Vietnam*
[2]*Phenikaa Research and Technology Institute (PRATI), A&A Green Phoenix Group JSC, No. 167 Hoang Ngan, Trung Hoa, Cau Giay, Hanoi 11313, Vietnam*

Correspondence should be addressed to Tien-Thinh Le; thinh.letien@phenikaa-uni.edu.vn

In this study, a hybrid machine learning (ML) technique was proposed to predict the bearing capacity of elliptical CFST columns under axial load. The proposed model was Adaptive Neurofuzzy Inference System (ANFIS) combined with Real Coded Genetic Algorithm (RCGA), denoted as RCGA-ANFIS. The evaluation of the model was performed using the coefficient of determination (R^2) and root mean square error (RMSE). The results showed that the RCGA-ANFIS ($R^2 = 0.974$) was more reliable and effective than conventional gradient descent (GD) technique ($R^2 = 0.952$). The accuracy of the present work was found superior to the results published in the literature ($R^2 = 0.776$ or 0.768) when predicting the load capacity of elliptical CFST columns. Finally, sensitivity analysis showed that the thickness of the steel tube and the minor axis length of the elliptical cross section were the most influential parameters. For practical application, a Graphical User Interface (GUI) was developed in MATLAB for researchers and engineers and to support the teaching and interpretation of the axial behavior of CFST columns.

1. Introduction

In recent decades, composite concrete-filled steel tubular (CFST) columns are considerably employed in the construction of infrastructures thanks to their excellent structural behavior [1]. These structural members exhibit many benefits than single material columns (i.e., concrete columns or hollow steel columns). These advantages could be listed as fire, axial capacity, and earthquake resistance [2, 3]. In practical engineering, various cross section geometries of CFST columns have been considered, such as circular [4], square [5], or rectangular cross sections [6]. Recently, the elliptical cross section was adopted in several works [3, 7, 8]. Indeed, the use of elliptical CFST columns has gained attention from the scientific community and applied engineering as it provides specific advantages compared to other cross sections of CFST, including a better strength and rigidity as well as fire resistance [9]. Due to its reasonable

distribution of the major-minor axis, elliptical CFST column exhibits a better architectural aesthetic appearance and a small fluid resistance coefficient [10, 11]. Moreover, the prevention of local buckling in the elliptical CFST columns could be well-established thanks to the concrete core [12, 13]. The elliptical section possesses aesthetic qualities along with more effective bending resistance when compared to circular section due to having different second moments of area around its principal axes [14]. Therefore, analyzing the structural behavior, especially the ultimate load of elliptical CFST columns, is essential to facilitate the use in civil engineering structures.

However, there are currently no standards or codes in any countries for assessing the load-carrying capacity of elliptical CFST columns [15]. Besides, there were several empirical formulations in the available literature such as Liu and Zha 2011 [16] and Shen et al. [17] for predicting the ultimate load of elliptical CFST members. However,

these equations were derived using assumptions and experimental observations, which led to a simplification of the prediction model. Consequently, the application of these models could not be extended to other results. All these limit the application of elliptical CFST columns in engineering practice. Although previous studies provided significant contributions to the progress in modeling and prediction of axial behavior of CFST members, a more robust and efficient model should be developed to reduce the cost and time consumed in experiments and field works.

Recently, machine learning (ML) approaches have been employed in various mechanical and civil engineering applications [18, 19], particularly for structural members under compression [20, 21]. As an example, Sarir et al. [22] proposed a tree-based and whale optimization model for predicting the load-bearing capacity of circular CFST members. Besides, Ahmadi et al. [23, 24] applied an artificial neural network for predicting the axial capacity of circular CFST short columns. In another work, Tran et al. [25] developed a neural network-based model for predicting the load-bearing capacity of square CFST columns. The obtained results in the literature demonstrated that ML methods have a very promising potential for predicting the mechanical behavior of structural elements. Despite the importance of elliptical CFST columns, most ML-based studies focused on circular and square cross sections [22, 26, 27]. Therefore, more investigations should be carried out to assess the potential applications of ML-based models for studying the axial behavior of elliptical CFST columns.

Therefore, the primary objective of the present work was to develop an ML-based model to predict the ultimate load of elliptical CFST columns under axial loading. For this purpose, a hybrid ML model, namely Adaptive Neurofuzzy Inference System (ANFIS) combined with Real Coded Genetic Algorithm (RCGA), was developed. The RCGA was chosen because of its higher optimization capability than the conventional gradient descent (GD) technique, as highlighted in this study. As the present work mainly focused on elliptical CFST columns, the input data included the length of the column, the major and minor axis lengths of the elliptical cross section, the thickness of the steel tube, and the mechanical properties of steel and concrete (i.e., yield strength and compressive strength, respectively). In order to train and validate the developed hybrid ML model, statistical quality assessments such as coefficient of determination (R^2) and root mean squared error (RMSE) were employed. Monte Carlo simulations were also carried out in order to estimate the robustness of the proposed ML model. A sensitivity analysis was conducted to investigate the influence of input variables on the prediction results. The prediction capacity of the RCGA-ANFIS model was also compared with existing equations in the literature for estimating the ultimate load of elliptical CFST columns. Finally, a Graphical User Interface (GUI) based on the developed ML model was provided, aiming at quick and efficient estimation of the ultimate load of elliptical CFST columns.

2. Materials and Methods

2.1. Database. In this work, a database was constructed by extracting available datasets from experimental research of Uenaka [28], Yang et al. [29], Liu et al. [30], Ren et al. [12], Dai et al. [31], Jamaluddin et al. [32], Yang et al. [33], McCann et al. [34], and Zhao and Packer [35]. From these investigations, a total number of 94 configurations were collected and summarized (Table 1), including the number of data points and proportion (in %). As revealed in the literature, the experimental procedure was conducted following the steps below:

  (i) Design of specimens

 (ii) Manufacturing of steel tube

(iii) Manufacturing of concrete core

(iv) Assembly of composite columns

 (v) Loading and measurement (see Figure 1 for a schematic description of the test as well as geometrical parameters of the members)

In terms of the experimental studies, various geometrical parameters, as well as mechanical properties of the constituent materials, were considered in order to test the failure of elliptical CFST columns under axial compression. For that reason, the input parameters of the problem regarding the geometry were the length of the column (denoted by L), the major axis length of the elliptical cross section (denoted by D), the minor axis length of the elliptical cross section (denoted by d), and the thickness of the steel tube (denoted by δ). Regarding the mechanical properties of constituent materials, the yield strength of the steel tube (denoted by f_y) and the compressive strength of the filled concrete (denoted by f_c') were considered. The ultimate load of the column under axial compression was the output of the problem, denoted by Q_n. A primarily statistical analysis of the database is indicated in Table 2, including the min, average, max, standard deviation (StD), and coefficient of variation (CV) values of all variables. It should be noticed that several statistical correlation techniques such as Principal Component Analysis [36] were applied, and no significant correlations were found in the input space. This confirmed that, for the prediction problem, all input parameters in this study were independent, and the selection of inputs was relevant. Finally, all data were scaled into the range of [−1, 1] in order to minimize numerical bias in the training phase.

2.2. Methods Used

2.2.1. Adaptive Neurofuzzy Inference System (ANFIS). The Adaptive Neurofuzzy Inference System, referred to as ANFIS, is an ML model constructed from the combination between a set of fuzzy if-then rules and the fuzzy inference systems through an adaptive network [37, 38]. The main idea of ANFIS is to construct a set of fuzzy if-then rules, including suitable membership functions to create the stipulated output and input variables [39, 40]. Supposing that the ANFIS model has two input variables such as X and Y and

TABLE 1: Organization of database.

Source of data	Number of data points	Proportion of data (%)
Uenaka [28]	19	20.2
Yang et al. [29]	2	2.1
Liu et al. [30]	18	19.1
Ren et al. [12]	6	6.4
Dai et al. [31]	13	13.8
Jamaluddin et al. [32]	17	18.1
Yang et al. [33]	9	9.6
McCann et al. [34]	2	2.1
Zhao and Packer [35]	8	8.5
Total	94	100

one output variable such as Z, we apply the following Takagi and Sugeno's if-then rules [41, 42]:

If X is A_1 and Y is B_1, then $Z_1 = a_1 X + b_1 Y + c_1$ (rule 1);

If X is A_2 and Y is B_2, then $Z_2 = a_2 X + b_2 Y + c_2$ (rule 2). (1)

Here, A and B are linguistic labels characterized by appropriate membership functions, and a, b, and c are the linear output parameters.

Consider the above ANFIS model with two input variables X and Y. Its structure can be divided into five main layers as follows [43]:

Layer 1: each node in this layer corresponds to a node function, which can be chosen to be bell-shaped with a minimum value equal to 0 and a maximum value equal to 1, for example, the Gaussian function, such that

$$\mu A_i(x) = \exp\left[-\left(\frac{x - a_i}{b_i}\right)^2\right],$$ (2)

where x is problem input and a_i, b_i are input parameters.

In fact, any continuous and differentiable functions can be chosen for the nodes in this layer.

Layer 2: each node in this layer is a node function that multiplies the incoming inputs and sends the results to the next layer:

$$w_i = \mu C_i^1(x_1) \times \mu C_i^2(x_2) \times \cdots \times \mu C_i^n(x_n).$$ (3)

Layer 3: each node in this layer computes the ratio between the ith rule's firing strength and the sum of all rules' firing strength:

$$\overline{w}_i = \frac{w_i}{\sum_{k=1}^n w_k}.$$ (4)

Layer 4: each node in this layer is a node function chosen such that

$$f_i = \overline{w}_i\left(c_0 + \sum_{k=1}^n c_k X_k\right).$$ (5)

Layer 5: the circle node in this layer calculates the sum of all incoming results and exports as the overall output

$$\text{Overall ouput} = \sum_i \overline{w}_i f_i.$$ (6)

The training algorithm uses a combination of the least-squares and backpropagation gradient descent methods to model the training dataset [44].

2.2.2. Real Coded Genetic Optimization Algorithm. Real Coded Genetic Algorithm, referred to as RCGA, is a metaheuristic optimization technique which is inspired by the principles of biological evolution. The basic idea of RCGA is to move a population of chromosomes, which are composed of strings of ones and zeros (or genes), to a new one that performs better than the old one [45]. There are two primary operations in RCGA, which are crossover and mutation [46, 47]. Crossover is a phase where the chromosomes in the population randomly share their features. This is the most significant operation in the RCGA, as more powerful offspring are created taking useful features from their parent's genes. Mutation is a process that is operated within each offspring, meaning that some of the bits in the bit string can be flipped. The main objective of the mutation process is to maintain the diversity of the population after new offspring are created from crossover [48].

The RCGA can be divided into five main steps as follows [48, 49]:

(i) Initial population. In this step, a set of chromosomes called population is defined. Each individual of the population corresponds to a solution of the considered problem. Each chromosome is formed by joining genes into a string. Typically, chromosomes are composed of strings of ones and zeros.

(ii) Fitness function. In this step, the fitness score of each individual in the population is calculated. It defines how to fit the chromosome or the ability of that chromosome to compete with others. A higher fitness score means that the individual is more likely to be reproduced.

(iii) Selection. In this step, the chromosomes with the highest values of fitness score will be selected in order to share their features in the next step.

(iv) Crossover. In this step, the crossover process will be operated on the most fitting chromosomes. Their genes are randomly exchanged to create new offspring.
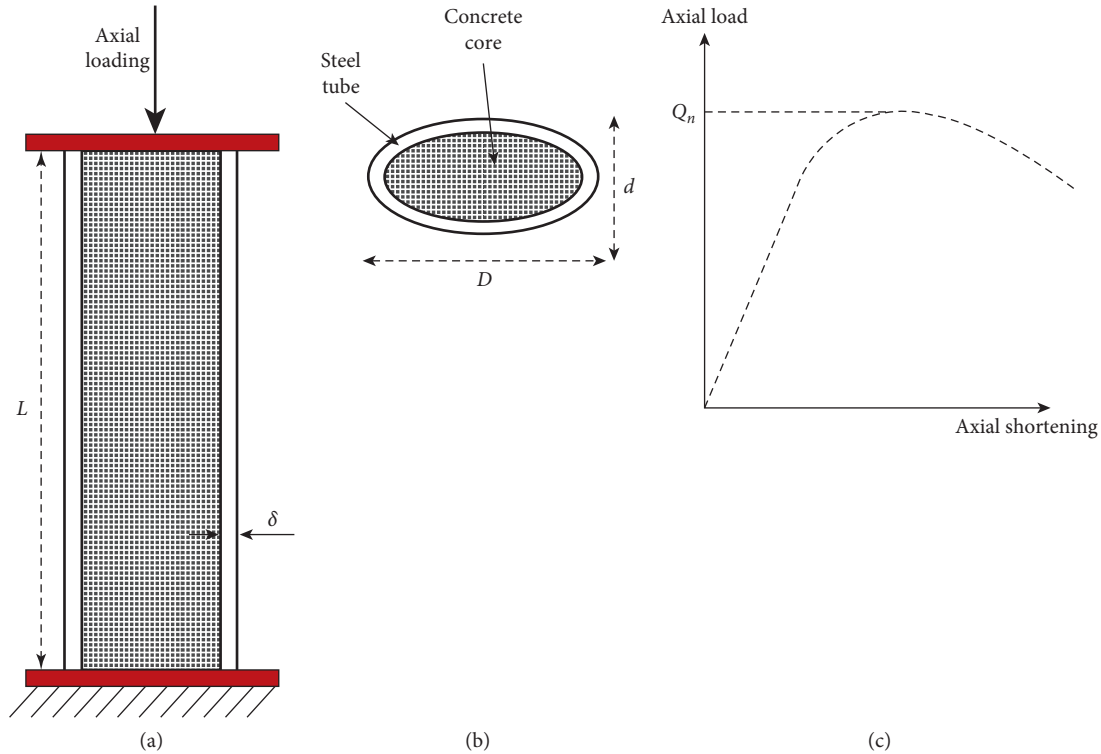
FIGURE 1: Schematization for (a) the CFST columns under axial loading, (b) the elliptical cross section, and (c) the load-axial shortening curve (a drawing based on experimental curves of Uenaka [28]).

TABLE 2: Initial statistical analysis of the database.

Parameter	Unit	Notation	Min	Average	Max	StD	CV (%)
Length of column	mm	L	160	991.86	3600	923.908	93.1
Major axis length of cross section	mm	D	136.5	177.281	318.5	35.986	20.3
Minor axis length of cross section	mm	d	63.1	93.693	155	21.466	22.9
Thickness of steel tube	mm	$\delta m1$	3.854	9.72	1.679	43.6	
Yield strength of steel tube	MPa	f_y	201	360.657	439.3	59.378	16.5
Compressive strength of concrete	MPa	f_c'	13.18	48.638	102.26	20.843	42.9
Ultimate load	kN	Q_n	413.3	1130.462	2607	484.164	42.8

(v) Mutation. In this step, the mutation process is done within each individual offspring to maintain the diversity of the population.

The algorithm is terminated when the model has converged, meaning that the newly created offspring are not different from the previous ones. In the literature, RCGA was used mainly in hybrid ML approaches [49]. For instance, Kim and Shin [50] used a hybrid approach based on neural networks and genetic algorithms for detecting temporal patterns, Le et al. [51] in steel structures applications, or Yan et al. [52] for engineering design problems. Finally, a complete review of the RCGA technique could be found in Lee [53].

2.2.3. Random Sampling Technique: Monte Carlo Method. The main idea of the Monte Carlo method is that the output is computed by repeating the sampling of variables randomly from the input space [54–56]. That way, (i) the Monte Carlo method is widely applied in order to propagate the variability of inputs on the output response; (ii) based on statistical analysis of output, several posttreatments such as robustness and/or sensitivity analyses could be thoroughly achieved [57] (see Figure 2 for a typical statistical problem using the Monte Carlo method). As shown in Figure 2, each input exhibits a probability distribution describing its variability. Due to the variabilities of input variables, the response also exhibits its statistical behaviors, which are necessary to be characterized [58]. The robustness of the model and/or sensitivity of input variables could then be deduced based on statistical analysis of output response [59–61].

Using Monte Carlo simulation, the bigger the number of realizations, the higher the reliability of the response archived. In this work, in order to optimize the number of Monte Carlo runs, a statistical estimator of convergence was applied, such as [62–65]
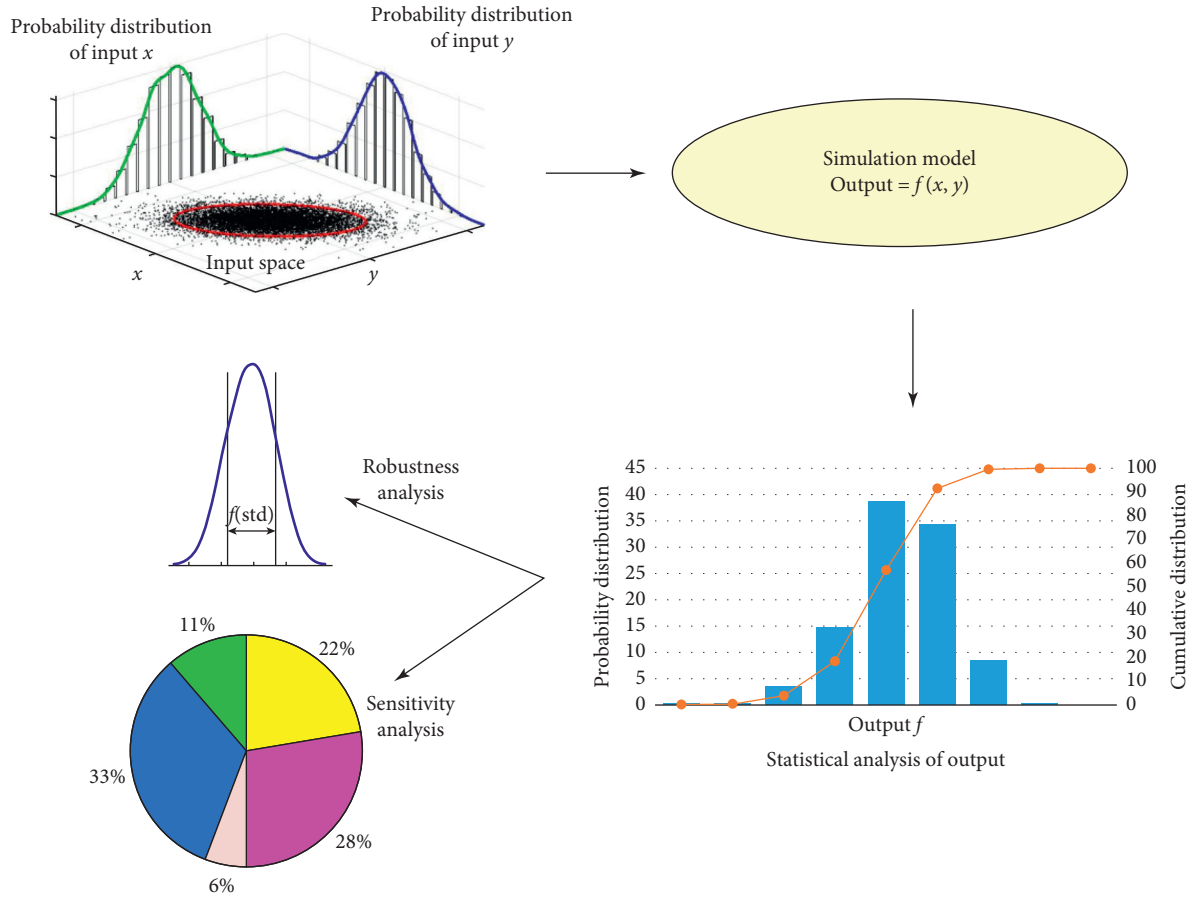
FIGURE 2: Monte Carlo simulation taking into account variability in the input space for robustness and sensitivity analysis.

$$N_{\text{MC}} \mapsto I\left(N_{\text{MC}}\right) = \frac{1}{\overline{W}} \frac{1}{N_{\text{MC}}} \sum_{i=1}^{N_{\text{MC}}} W_i, \qquad (7)$$

where $\overline{W}$ is the mean of the considered variable W and N_{MC} is the number of Monte Carlo runs.

2.3. Quality Assessment Criteria.
In the present work, statistical criteria, namely, the coefficient of determination (R^2) and Root Mean Squared Error (RMSE), have been used in order to validate and test the developed ML model. The R^2 allows us to identify the statistical relationship between two data points. This measurement of the linear correlation yields a value between 0 and 1 inclusively, where 0 is no correlation and 1 is total correlation. R^2 could be calculated using the following equation [66, 67]:

$$R^2 = \frac{\left(\sum_{k=1}^{N} \left(p_k - \overline{p}\right)\left(w_k - \overline{w}\right)\right)^2}{\sum_{k=1}^{N} \left(p_k - \overline{p}\right)^2 \sum_{k=1}^{N} \left(w_k - \overline{w}\right)^2}, \qquad (8)$$

where N is the number of the observations, p_k and $\overline{p}$ are predicted and mean predicted values, and w_k and $\overline{w}$ are measured and mean measured values of ultimate load, respectively ($k = 1: N$). The formulation of RMSE is described by the following equation [68–70]:

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{k=1}^{N} \left(p_k - w_k\right)^2}. \qquad (9)$$

Finally, the slope criterion is defined, such as the slope of the linear regression fit between predicted and observed vectors.

3. Results and Discussion

3.1. Optimization of ANFIS's Weight Parameters.
In this section, the optimization of ANFIS's weight parameters is presented. Such optimization procedure was done using both conventional GD and advanced RCGA techniques, respectively, to identify the best training algorithm. Table 3 indicates the characteristics of ANFIS, including the type of membership function, the number of weights per membership function, and the number of membership functions per input as well as the number of nodes. It is seen that there were 190 consequent and antecedent ANFIS parameters to be optimized as ANFIS was generated using the c-means clustering algorithm for the considered six-dimensional input space [71, 72]. In this study, a maximum number of iterations of 1000 was employed as the stopping condition when optimizing. The cost function was selected as RMSE.

TABLE 3: Parameters of ANFIS and RCGA used in this study.

Parameter of ANFIS	Value	Parameter of RCGA	Value
Number of inputs	6	Population size	100
Number of outputs	1	Length of chromosome	190
Membership function	Gaussian	Fitness function	Linear ranking
Number of parameters per membership function	2	Crossover type	Random pair
Number of membership functions per input (rules)	10	Crossover probability	0.4
Number of nodes	149	Number of offsprings	12
Number of nonlinear parameters of the antecedent membership function	120	Mutation type	Random
Number of linear parameters of the consequent membership function	70	Mutation probability	0.7
Total number of parameters	190	Number of mutants	21
Cost function	RMSE	Mutation rate	0.15
		Selection function	Fitness proportionate selection

The parameters of RCGA during the training phase are also indicated in Table 3. Figures 3(a) and 3(c) present the evolution of RMSE during the optimization process, using GD and RCGA, respectively. The same illustration is presented in Figures 3(b) and 3(d), but for the evolution of R^2. It should be noticed that, in these figures, the value of RMSE (i.e., R^2) for training and testing data was also highlighted during the learning phase. It is seen that at least 600 iterations were needed for obtaining a convergence with respect to both RMSE and R^2. At the same time, the evolution of RMSE and R^2 is plotted using the testing data, which were totally new when applied. Such evolution exhibits efficiency during the training process; i.e., no overfitting or underfitting was observed.

The values of all quality assessment criteria at the end of the training process are indicated in Table 4, whereas the results in terms of regression plots and error distribution are shown in Figures 4(a)–4(c), respectively. As indicated in Table 4, using the training data, RCGA-ANFIS provided the highest value of R^2, which is 0.971, while the R^2 value of GD-ANFIS is 0.933. In terms of RMSE, RCGA-ANFIS yielded the smallest value, which is 70.379 kN, whereas the RMSE value of GD-ANFIS is 105.428 kN. In terms of linear fit, the RCGA-ANFIS model produced the highest value of slope (0.98) corresponding to a slope angle of 44.425°, while the slope value of GD-ANFIS was 0.937 corresponding to slope angle of 43.125°. Regarding error analyses, using the training data, the mean values are 1.409 and 0.972%, while the corresponding standard deviation values are 11.082 and 8.497% for GD-ANFIS and RCGA-ANFIS, respectively. It can be seen that the RCGA-ANFIS model yielded an error mean, which is the closest to zero and the smallest standard deviation value (see also Figure 4(c)). The application of the two ML models to the validating data is presented in the next section.

3.2. Validation of Model. The previously developed GD-ANFIS and RCGA-ANFIS models were applied to the validating data for validation. As a result, Figures 5(a) and 5(b) present regression graphs between actual and predicted ultimate load, whereas Figure 5(c)shows error distribution,

respectively. All quantitative values of quality assessment criteria are indicated in Table 4. As indicated in Table 4, using the validating data, GD-ANFIS provided $R^2 = 0.952$, RMSE = 130.065 kN, Mean$_{error}$ = −0.456 kN, StD$_{error}$ = 8.967 kN, and slope = 0.920, whereas RCGA-ANFIS provided $R^2 = 0.974$, RMSE = 100.340 kN, Mean$_{error}$ = 2.541 kN, StD$_{error}$ = 8.042 kN, and slope = 1.019, respectively. The same remarks were obtained for the training data, RCGA-ANFIS yielded the best prediction performance. It could be stated that the RCGA-ANFIS model is validated because it performs well the prediction of ultimate load using the validating data. Thus, RCGA-ANFIS model was selected as the final prediction model for estimating the ultimate load of elliptical CFST columns.

3.3. Sensitivity Analysis. In this section, the influence of input variables on the prediction of column load-carrying capacity is presented. For this purpose, the probability distribution of each input was characterized by 11 levels of quantiles such as Q_0, Q_{10}, Q_{20}, Q_{30}, Q_{40}, Q_{50}, Q_{60}, Q_{70}, Q_{80}, Q_{90}, and Q_{100}. For a given input, a local influence index, denoted by θ (in %), was computed by the following equation:

$$\theta_q^k = \frac{\left| Q_q^k - Q_{median}^{all} \right|}{Q_{median}^{all}} \times 100, \tag{10}$$

where Q_{median}^{all} is the output, the ultimate load when all inputs are equal to their Q_{50} values. Q_q^k is the output of the ML model when applying kth input at its qth levels (quantiles from 0 to 100 every 10, respectively) ($k = 1, \ldots, 6$ and $q = 1, \ldots, 11$). That way, the global influence index of the kth input, denoted by M^k, is calculated as follows:

$$M^k = \sum_{q=1}^{11} \theta_q^k. \tag{11}$$

Figures 6(a) and 6(b) present the global influence index of all inputs parameters using GD-ANFIS and RCGA-ANFIS, respectively (see the appendix for statistical convergence of Monte Carlo simulations). It could also be
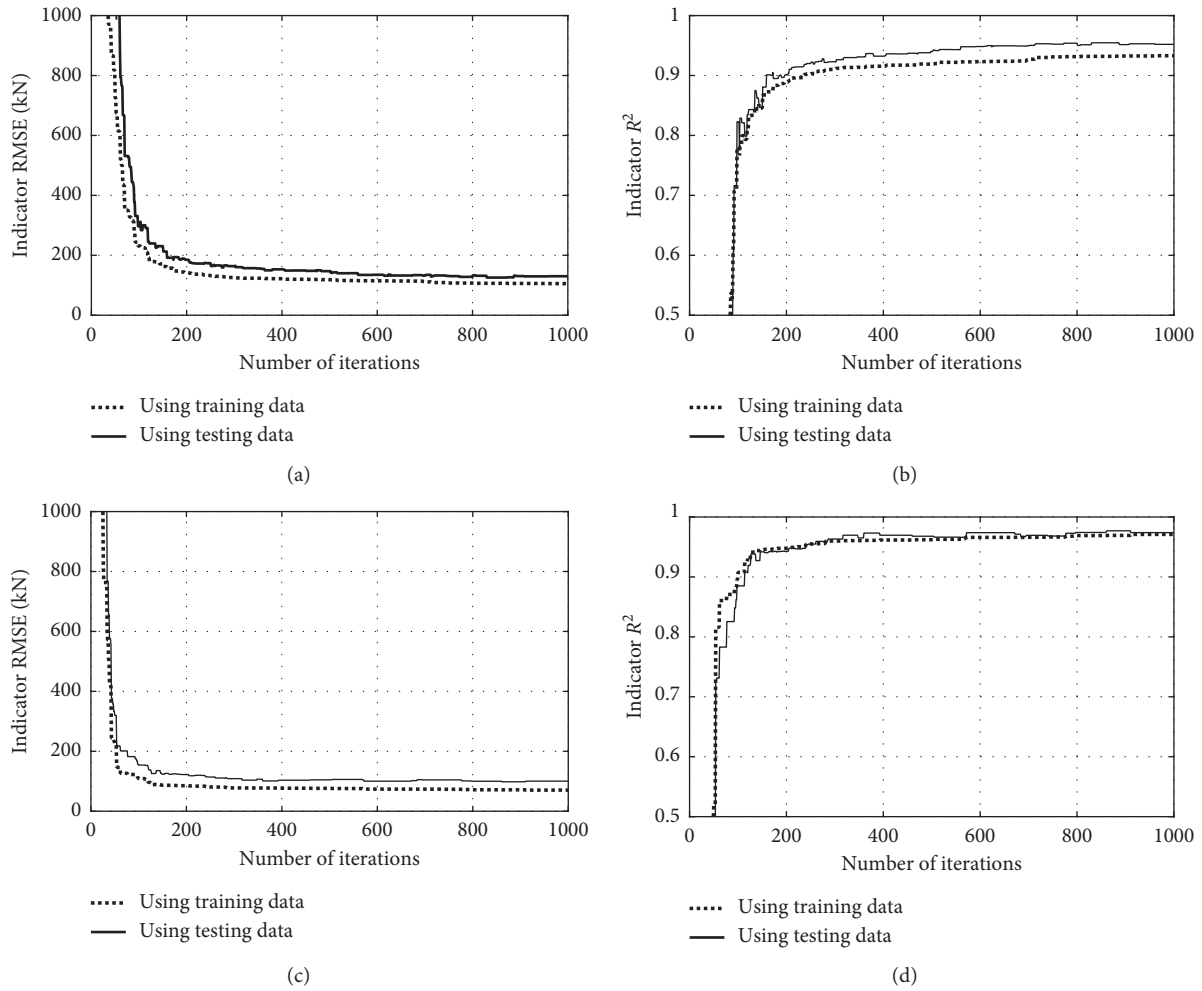
FIGURE 3: Evolution during optimization process for RMSE using (a) GD and (c) RCGA; for R^2 using (b) GD and (d) RCGA.

TABLE 4: Summary of prediction capability.

Data used	Model	R^2	RMSE	Mean$_{error}$ (%)	StD$_{error}$ (%)	Slope	Slope angle (°)
Training	GD-ANFIS	0.933	105.428	1.409	11.082	0.937	43.125
	RCGA-ANFIS	0.971	70.397	0.972	8.497	0.980	44.425
Testing	GD-ANFIS	0.952	130.065	−0.456	8.967	0.920	42.600
	RCGA-ANFIS	0.974	100.340	2.541	8.042	1.019	45.536

noticed that the bar graphs are reorganized in decreasing order of the mean value for all six input variables. All values are indicated in Table 5. It is clearly observed that all input variables affect the axial capacity of structural members considerably under axial compression from a minimum of 6.1% to a maximum of 22.5% on average. It is also seen that the axial capacity is in function of inputs under a nonlinear form (i.e., a linear equation could not join all mean values of sensitivity index). It is seen that there are at least four levels of influence ranking. Indeed, the two most important variables are d and δ, which exhibit more than 20% of influence each. Next, L and D could be classified in the second group, which exhibit about 18% of influence each. The third group

contains the compressive strength of concrete, whereas the yield strength of steel has about 6% of influence and is in the last group. Last but not least, it is seen that the fluctuation of the influence index obtained by GD-ANFIS is higher than the ones obtained by RCGA-ANFIS. This points out that RCGA-ANFIS is more robust and efficient than GD-ANFIS, which confirms the higher performance of RCGA than GD, as identified in Section 3.2.

3.4. Comparison with Existing Models. In this section, the best prediction model, namely RCGA-ANFIS, is compared with existing models in the literature for the axial capacity of
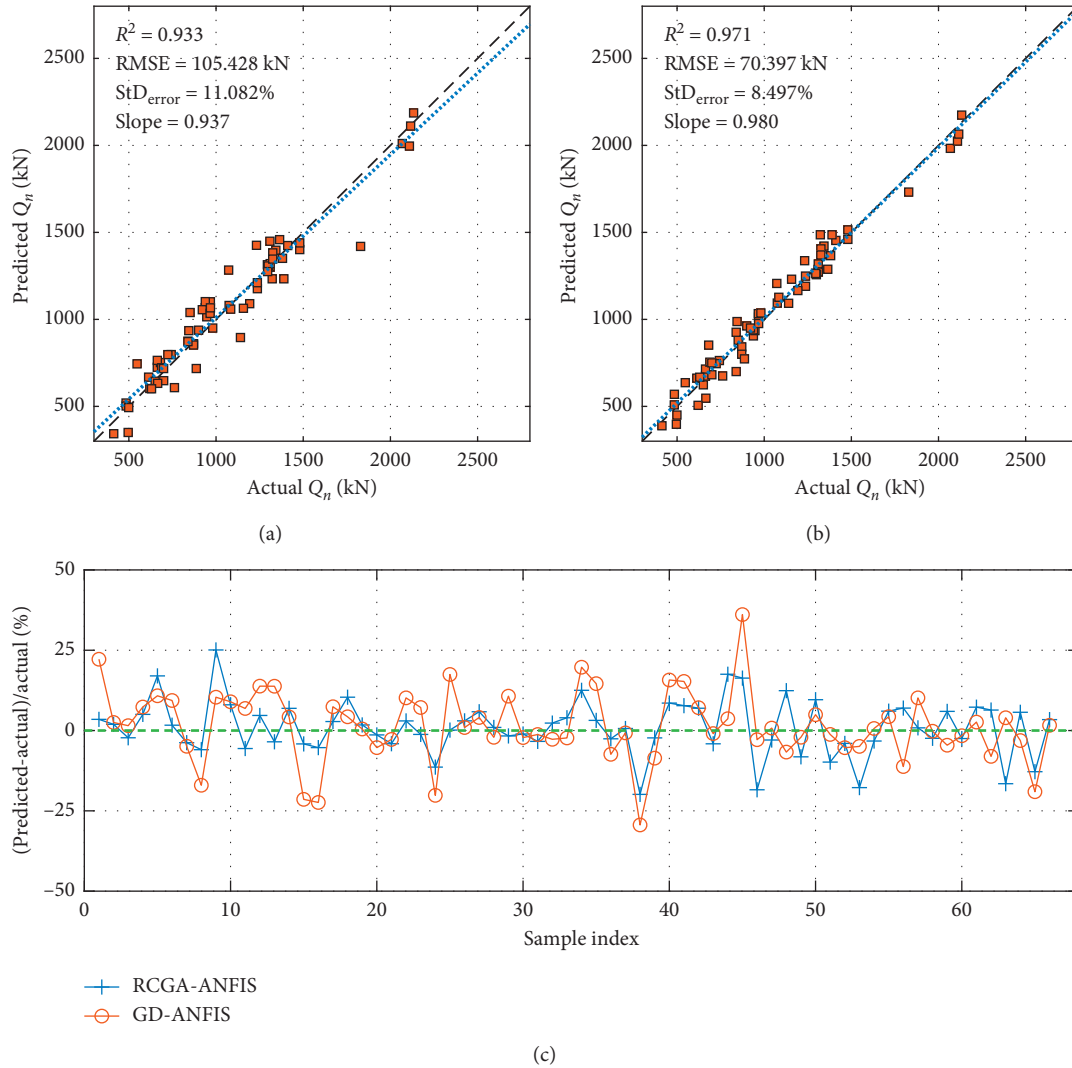
FIGURE 4: Results after training process for (a) using GD, (b) using RCGA, and (c) distribution of errors.

elliptical CFST columns. Liu and Zha [16] have proposed the following equation:

$$Q_n^{\text{Liu}-2011} = \frac{1 + 1.5\,(d/D)^{0.3}}{1 + (A_s/A_c)} \times \frac{A_s f_y}{A_c}, \tag{12}$$

where A_s and A_c are the cross-sectional area of the steel tubular and the concrete core, respectively. Another formula for predicting the axial capacity of elliptical CFST columns was developed by Shen et al. [17], such as

$$Q_n^{\text{Shen}-2015} = f_c'\,(A_s + A_c)\left[0.0075 \times \left(\frac{A_s f_y}{A_c f_c'}\right)^3 + 0.0624 \times \left(\frac{A_s f_y}{A_c f_c'}\right)^2 + 0.7080 \times \left(\frac{A_s f_y}{A_c f_c'}\right) + 1.3625\right]. \tag{13}$$

Figures 7(a)–7(c) present the regression graph between actual and predicted ultimate load, using Liu et al. 2011, Shen et al. 2015, and RCGA-ANFIS model, respectively. All

performance indicators are also highlighted in Table 6. It is seen in Figure 7 and Table 6 that the RCGA-ANFIS model provided better performance than the literature, with respect
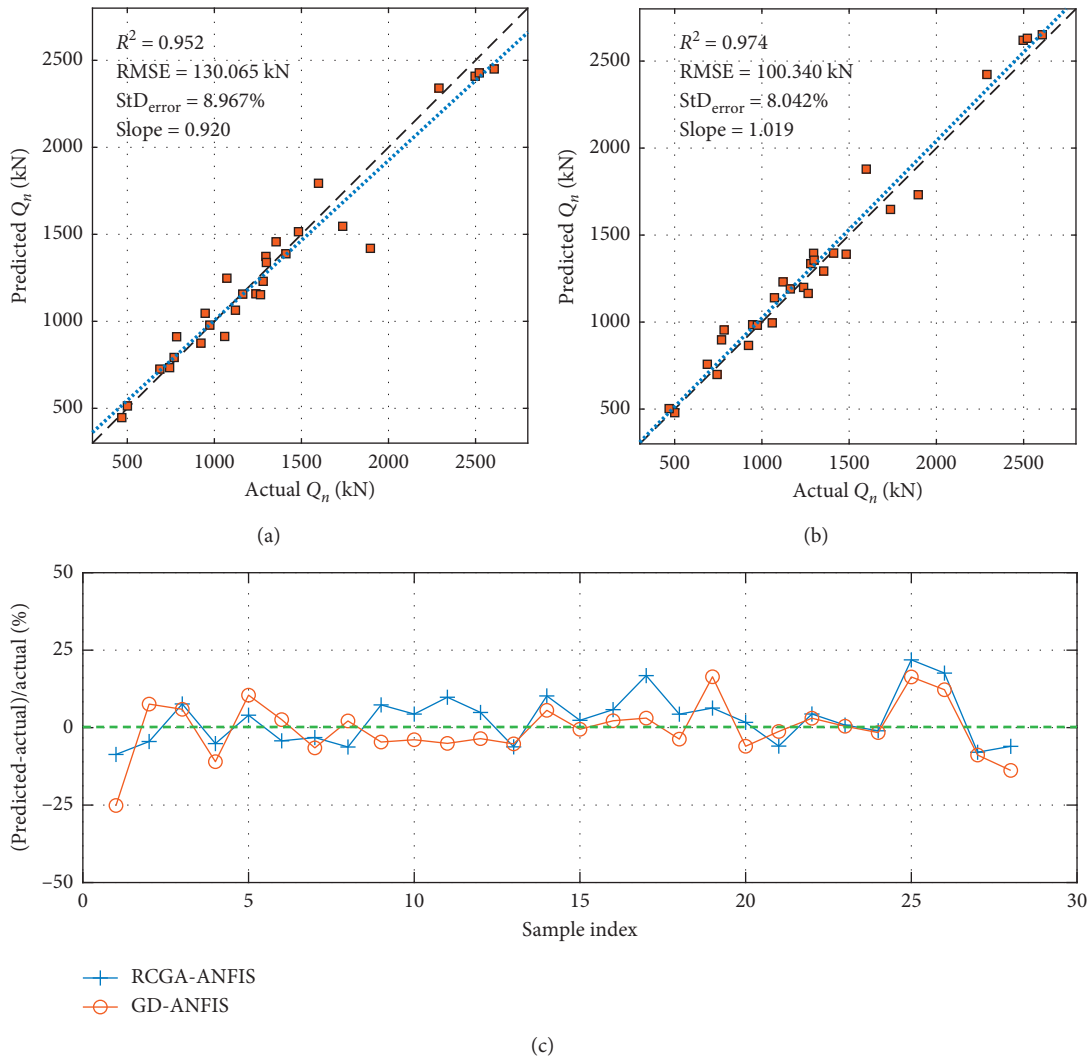
Figure 5: Results after validating process for (a) using GD, (b) using RCGA, and (c) distribution of errors.
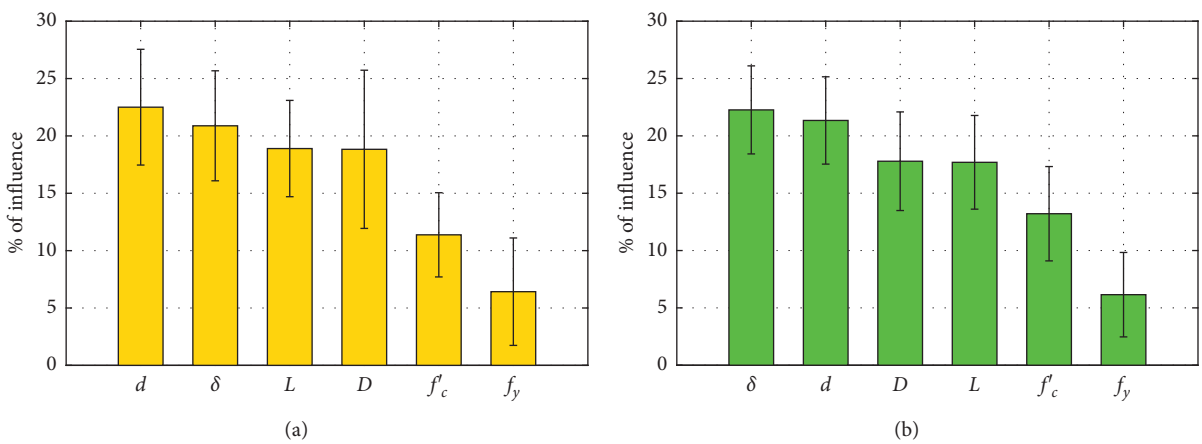
Figure 6: Sensitivity analysis of input variables using (a) GD-ANFIS and (b) RCGA-ANFIS.

TABLE 5: Statistical analysis of global influence index (in %).

Parameter	Model	L	D	d	δ	f_y	f_c'
Mean	GD-ANFIS	18.898	18.832	22.505	20.882	6.422	11.378
	RCGA-ANFIS	17.692	17.788	21.344	22.264	6.151	13.210
StD	GD-ANFIS	4.198	6.895	5.045	4.795	4.683	3.671
	RCGA-ANFIS	4.085	4.300	3.807	3.839	3.682	4.113

(a)

(b)

(c)
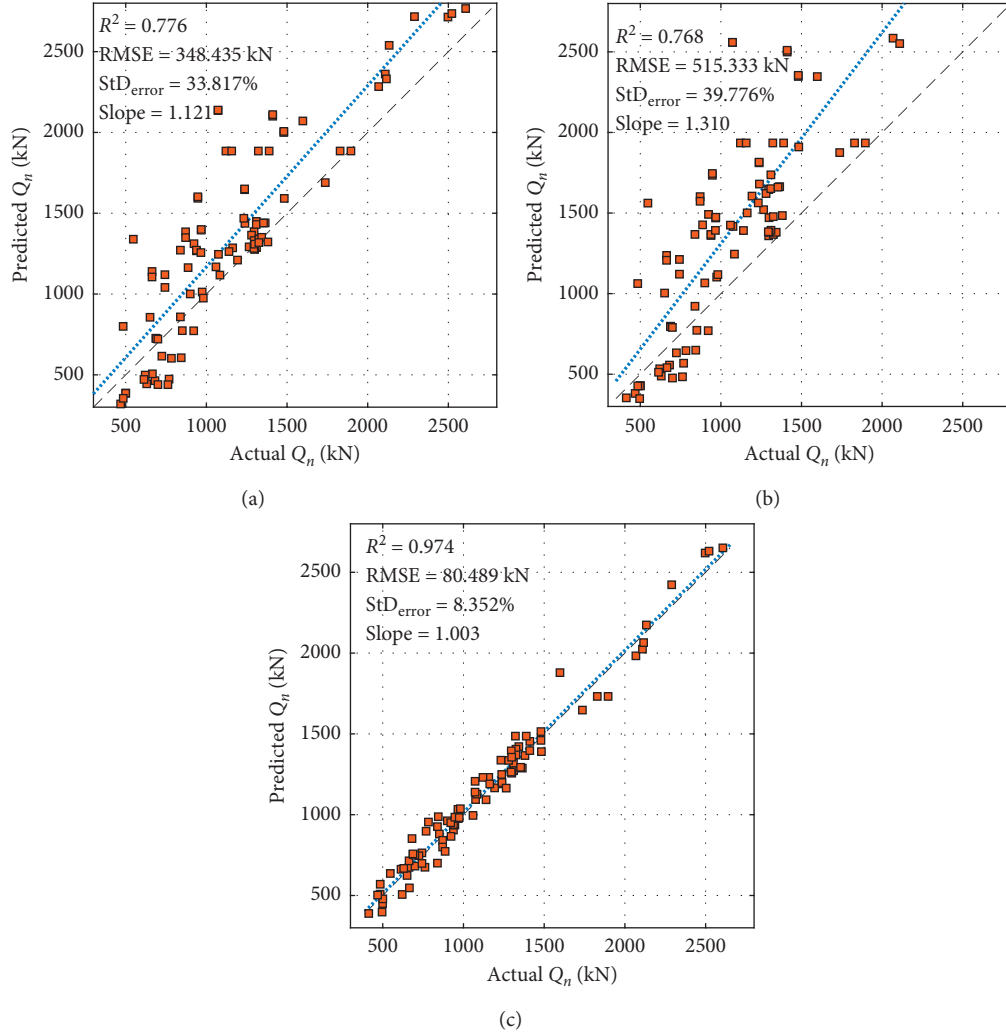
FIGURE 7: Regression graphs between predicted and actual Q_n (all data) using (a) Liu and Zha [16], (b) Shen et al. [17], and (c) RCGA-ANFIS model.

to all error measurement criteria. In Table 6, the percentage of gain is also indicated. The percentage of gain is calculated based on the following equation:

$$
\% \, \text{Gain} = \begin{cases} \left(\left(\Omega^{\text{this-study}} - 1 \right) - \left(\Omega^{\text{literature}} - 1 \right) \right) \times 100, & \text{in case of: } R^2 \text{ and Slope;} \\[2em] \left(\dfrac{\left(\Omega^{\text{literature}} - \Omega^{\text{this-study}} \right)}{\Omega^{\text{literature}}} \right) \times 100, & \text{in case of: RMSE and ErrorStD.} \end{cases}
\tag{14}
$$

TABLE 6: Comparison between RCGA-ANFIS model and literature.

Parameter	Model used	R^2	RMSE	$Mean_{error}$	StD_{error}	Slope	Slope angle (°)
	Liu et al. 2011	0.776	348.435	15.523	33.817	1.121	48.277
Performance indicator	Shen et al. 2015	0.768	515.333	29.887	39.776	1.310	52.636
	This work	0.974	80.489	1.439	8.352	1.003	45.092
% of gain	Liu and Zha [16]	+19.8	+76.9	+90.7	+75.3	+11.8	+7.1
	Shen et al. [17]	+20.6	+84.4	+95.2	+79.0	+30.6	+16.8

(a)

(b)

(c)
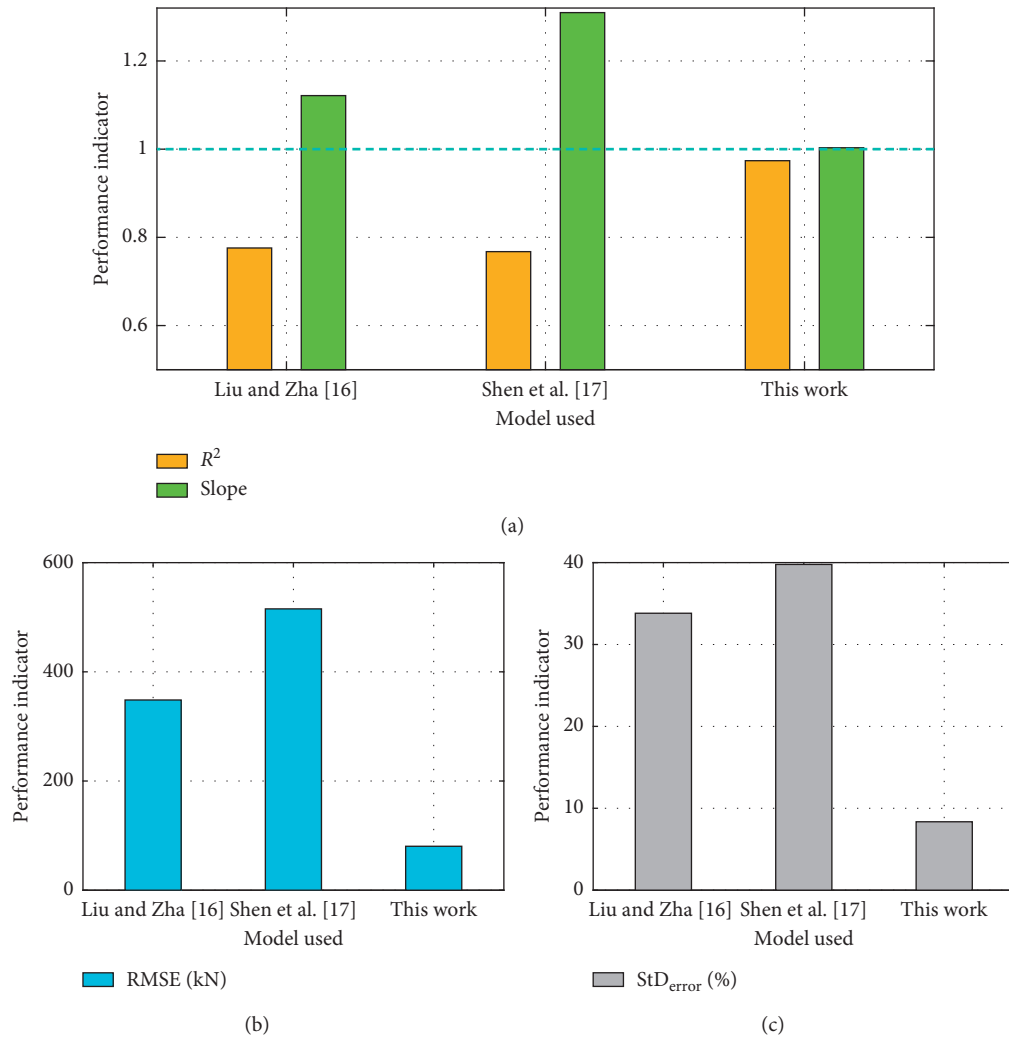
FIGURE 8: Comparison between RCGA-ANFIS model and literature: (a) in terms of R^2 and slope, (b) in terms of RMSE, and (c) in terms of StD_{error}.

Figure 8 shows the comparison regarding the performance indicators between RCGA-ANFIS and existing models. Obviously, the RCGA-ANFIS model showed an excellent performance in predicting the ultimate load of the elliptical CFST columns.

3.5. Practical Application.

For further application of RCGA-ANFIS model, a Graphical User Interface (GUI) was developed in MATLAB 2018a [73]. Figure 9 presents the main GUI, which is simple and easy to use. Users can enter the values of input variables; the ultimate load of elliptical CFST columns is then displayed directly by clicking the Start Predict button. The GUI is provided freely at https://github.com/Tien-ThinhLe/EllipticalCFST_AxialCapacityPrediction.

3.6. Proposed Empirical Formula.

It is not convenient for researchers/engineers to employ machine learning models in practice, because such a model contains weights, bias parameters, and transfer functions. Thus, an empirical formula based on the developed machine learning model should be derived to be employed in the engineering field. Based on the results obtained from the machine learning model, a mathematical method was used to derive a practical
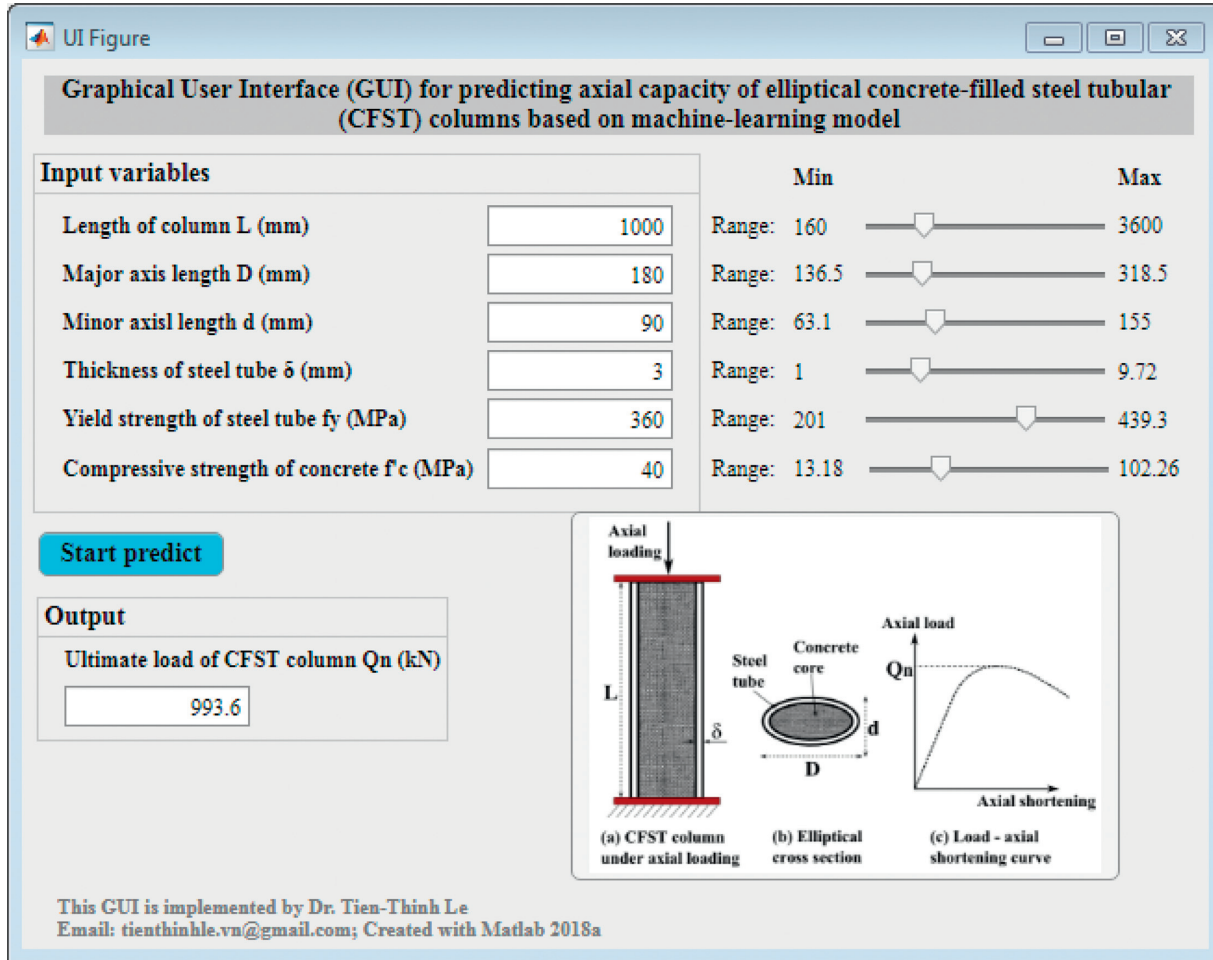
FIGURE 9: MATLAB's GUI for the prediction of the ultimate load of elliptical CFST columns based on RCGA-ANFIS model.

equation for the prediction of ultimate load of elliptical CFST columns. Such a procedure was inspired by a recent development of Nikbin et al. [74] in deriving an empirical formula for prediction of fracture energy of concrete based on machine learning models. Figure 10 presents the diagram of the procedure. More details could be found in Nikbin et al. [74].

Based on the procedure presented in Figure 10, the ultimate load of elliptical CFST columns can be predicted using

$$Q_n^{\text{Proposed formula}} = C_L \times C_D \times C_d \times C_\delta \times C_{f_y} \times C_{f_c'}, \quad (15)$$

where

$$C_L = -0.0075912 \times \left(\frac{L}{1000}\right)^2 - 0.15675 \times \left(\frac{L}{1000}\right) + 1.2827, \quad (16)$$

$$C_D = -0.25383 \times \left(\frac{D}{180}\right)^2 + 1.313 \times \left(\frac{D}{180}\right) - 0.016222, \quad (17)$$

$$C_d = 0.06122 \times d^2 - 2.7245 \times d + 501.50, \quad (18)$$

$$C_\delta = 0.11857 \times \left(\frac{\delta}{4}\right)^2 + 0.1051 \times \left(\frac{\delta}{4}\right) + 0.83073, \quad (19)$$

$$C_{f_y} = 0.51644 \times \left(\frac{f_y}{350}\right)^2 - 0.70249 \times \left(\frac{f_y}{350}\right) + 1.2644, \quad (20)$$

$$C_{f_c'} = 0.015364 \times \left(\frac{f_c'}{50}\right)^2 + 0.25698 \times \left(\frac{f_c'}{50}\right) + 0.80208. \quad (21)$$

The coefficients presented in (16)–(21) were deduced based on a least square optimization process (see also Nikbin et al. [74]). In order to evaluate the performance of the proposed equation, 94 experimental data points have been employed for a comparison purpose. Details of the experimental dataset, including input variables (geometric variables and strength of constituent materials), output variable (measured ultimate load), and three ratios $(Q_n^{\text{Liu}-2011}/Q_n)$, $(Q_n^{\text{Shen}-2015}/Q_n)$, $(Q_n^{\text{Proposed formula}}/Q_n)$, are indicated in Table 7. At the end of Table 7, statistics of the three ratios are also indicated, including the min, average, max, standard deviation, and coefficient of
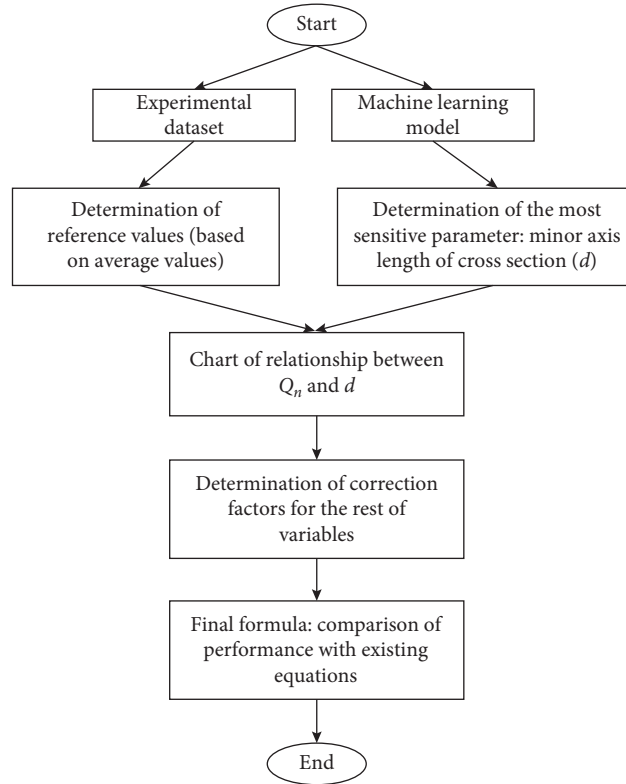
Figure 10: Methodology flowchart for the development of empirical formula.

Table 7: Comparison of performance between the proposed formula and existing equations.

L	D	d	δ	f_y	f_c'	Q_n	$(Q_n^{\text{Liu}-2011}/Q_n)$	$(Q_n^{\text{Shen}-2015}/Q_n)$	$(Q_n^{\text{Proposed formula}}/Q_n)$
mm	mm	mm	mm	MPa	MPa	kN	—	—	—
300	150.4	75.6	4.18	376.5	26.93	839	1.02	1.10	0.96
300	150.57	75.52	4.19	376.5	47.3	974	1.04	1.13	0.93
300	150.39	75.67	4.18	376.5	84.57	1265	1.02	1.20	0.86
300	150.12	75.65	5.12	369	26.93	981	0.99	1.14	0.88
300	150.23	75.74	5.08	369	47.3	1084	1.03	1.15	0.89
300	150.28	75.67	5.09	369	84.57	1296	1.07	1.27	0.90
300	148.78	75.45	6.32	400.5	26.93	1193	1.01	1.35	0.83
300	148.92	75.56	6.43	400.5	47.3	1280	1.07	1.27	0.88
300	149.53	75.35	6.25	400.5	84.57	1483	1.07	1.29	0.90
500	150.18	75.21	4.51	395	69.2	1075	1.16	1.32	0.96
500	150.49	75.26	5.41	358	69.2	1163	1.11	1.29	0.92
500	150.05	75.42	6.56	369	69.2	1310	1.11	1.33	0.92
600	200.21	100.12	5.2	397	69.2	1598	1.30	1.47	1.11
600	200	100.35	6.1	411	69.2	2068	1.10	1.25	0.95
600	200.6	100.02	8.17	383	69.2	2133	1.19	1.41	1.08
600	200.19	100.41	9.72	367	69.2	2290	1.19	1.46	1.15
698	220.7	110.7	6.16	421	48.2	2109	1.12	1.21	1.01
300	150.1	75	4.1	431.4	35.8	900	1.11	1.18	0.99
299	150.1	75.2	4.2	431.4	92.14	1239	1.16	1.36	0.97
398	197.8	100.1	5.1	347.9	36.87	1232	1.19	1.27	1.16
398	197.5	100.2	5.1	347.9	53.54	1737	0.97	1.08	0.90
398	197.4	100.1	5.1	347.9	102.26	2116	1.10	1.35	0.94
1497	150.9	75.4	4	431.4	17.9	650.8	1.32	1.54	1.03
1498	150.4	75.2	4.1	431.4	51.29	742.8	1.51	1.63	1.10
1496	150.3	75.2	4.1	431.4	77	923.2	1.42	1.62	1.01
1499	197.5	100.2	5.2	347.9	20.33	938.4	1.35	1.45	1.18

TABLE 7: Continued.

L	D	d	δ	f_y	f'_c	Q_n	$(Q_n^{\text{Liu}-2011}/Q_n)$	$(Q_n^{\text{Shen}-2015}/Q_n)$	$(Q_n^{\text{Proposed formula}}/Q_n)$
1498	197.7	100.1	5.1	347.9	77	1480	1.35	1.59	1.01
1785	150.7	75.2	4.2	431.4	51.67	663.2	1.72	1.87	1.18
1786	150.7	75.4	4.1	431.4	86.08	871.2	1.59	1.84	1.06
1785	197.6	100.2	5.1	347.9	31.32	967.5	1.44	1.52	1.15
1786	197.7	100.1	5.1	347.9	50.27	1237	1.33	1.46	1.00
1786	197.3	100	5.2	347.9	83.87	1411.2	1.49	1.77	1.05
2499	197.8	100.1	5.1	347.9	46.2	947	1.68	1.83	1.10
2498	197.7	100.1	5.1	347.9	87.28	1072.3	1.99	2.39	1.20
160	160	107.8	1	207	27.3	768.7	0.62	0.74	1.20
160	159.4	106.5	1.6	296	27.3	844	0.72	0.77	1.12
160	159.7	107.4	2.3	341	27.3	921.3	0.84	0.84	1.12
250	159.9	105.5	1	207	27.3	681.3	0.68	0.82	1.31
250	160.1	105.5	1.6	296	27.3	783.3	0.77	0.83	1.19
250	160.8	107	2.3	341	27.3	850.7	0.91	0.91	1.20
160	159.4	80.5	1.6	279	25	699.7	0.63	0.68	0.99
160	158.8	80.7	2.3	201	25	761.5	0.58	0.64	0.95
250	160.8	74.9	1	211	25	468.4	0.68	0.81	1.35
250	158.3	82	2.3	201	25	630.1	0.71	0.78	1.15
160	159.2	63.2	1	207	27.3	496	0.58	0.70	1.16
160	159.6	63.3	1.6	296	27.3	500.6	0.77	0.86	1.20
160	159.5	64.2	2.3	341	27.3	665.3	0.76	0.81	0.98
250	158.5	64.5	1	207	27.3	413.3	0.71	0.86	1.39
250	159.3	63.1	1.6	296	27.3	499.3	0.77	0.86	1.19
250	158.8	63.2	2.3	341	27.3	620.6	0.80	0.86	1.02
200	158.6	79.7	1	207	27.3	484.6	0.73	0.88	1.38
200	158	80.6	1.6	296	27.3	613.3	0.77	0.84	1.14
200	159.2	82.1	2.3	341	27.3	724	0.85	0.87	1.06
1678	150.4	75.2	4.1	410	45.64	743	1.40	1.51	1.00
1679	197.5	100	5.2	350	20.33	938	1.36	1.46	1.15
1678	197.7	100	5.1	350	77.2	1480	1.36	1.59	0.98
1965	150.5	75.4	4.1	410	13.18	484	1.65	2.20	1.21
1965	150.7	75.2	4.2	410	52.13	663	1.67	1.82	1.11
1966	150.7	75.4	4.1	410	86.18	871	1.55	1.81	1.00
1965	197.6	100	5.1	350	31.32	968	1.45	1.52	1.11
1966	197.7	100	5.1	350	50.27	1237	1.33	1.47	0.97
1966	197.3	100	5.2	350	84.17	1411	1.50	1.78	1.01
2681	150.1	75	4.1	410	86.18	547	2.45	2.86	1.35
2678	197.5	100	5.2	350	20.33	839	1.52	1.63	1.04
2679	197.8	100	5.1	350	46.56	947	1.69	1.84	1.06
2678	197.7	100	5.1	350	87.18	1072	1.99	2.39	1.15
3600	192	124	3.82	439.3	48.41	1121	1.68	1.73	0.89
3600	192	124	3.82	439.3	48.41	1157	1.63	1.67	0.86
2700	192	124	3.82	439.3	48.41	1389	1.36	1.39	0.93
2700	192	124	3.82	439.3	48.41	1322	1.43	1.46	0.98
1800	192	124	3.82	439.3	48.41	1896	0.99	1.02	0.83
1800	192	124	3.82	439.3	48.41	1829	1.03	1.06	0.86
2154	148.45	75.78	6.3	369.1	32	886.6	1.31	1.61	0.82
1154	148.37	75.63	6.3	369.1	33	1059.3	1.10	1.35	0.82
271	136.5	136.5	2.75	376.4	50.36	1296.3	1.03	1.05	1.16
271	137	137	2.75	376.4	50.36	1325.3	1.01	1.03	1.14
271	137.8	137.8	2.75	376.4	50.36	1343	1.01	1.03	1.14
338	170	112	2.75	376.4	50.36	1310.6	0.98	1.06	1.03
338	169.6	111	2.75	376.4	50.36	1299.2	0.98	1.06	1.03
338	168	112.5	2.75	376.4	50.36	1294.4	0.99	1.07	1.04
407	202	99	2.75	376.4	50.36	1298.7	1.01	1.13	1.01
407	199.8	100.8	2.75	376.4	50.36	1325	1.00	1.12	1.01
407	201.5	100.4	2.75	376.4	50.36	1381.1	0.96	1.07	0.97
475	236	95.8	2.75	376.4	50.36	1309.2	1.09	1.26	1.07
475	237.5	96	2.75	376.4	50.36	1364.6	1.06	1.22	1.03
475	236	96.5	2.75	376.4	50.36	1354.2	1.06	1.23	1.04

TABLE 7: Continued.

L	D	d	δ	f_y	f_c'	Q_n	$(Q_n^{Liu-2011}/Q_n)$	$(Q_n^{Shen-2015}/Q_n)$	$(Q_n^{Proposed\ formula}/Q_n)$
636	318	155	2.75	376.4	50.36	2607	1.06	1.24	1.21
636	318.5	151.5	2.75	376.4	50.36	2497.3	1.09	1.28	1.22
636	317	153.5	2.75	376.4	50.36	2521.5	1.08	1.27	1.23
279	139	68	2.75	376.4	50.36	687.2	1.06	1.16	1.04
279	138	68.2	2.75	376.4	50.36	688.1	1.05	1.15	1.04
279	137.5	68	2.75	376.4	50.36	699.2	1.03	1.13	1.02
2670.4	199.7	105.7	2.6	376.4	45	1140	1.11	1.22	0.79
1910.4	204.3	103.1	2.6	376.4	45	966	1.30	1.44	1.09
						Min	0.58	0.64	0.79
						Average	1.16	1.30	1.05
						Max	2.45	2.86	1.39
						StD*	0.34	0.40	0.13
						CV**	29.27	30.62	12.55

StD: standard deviation, CV: coefficient of variation (%).
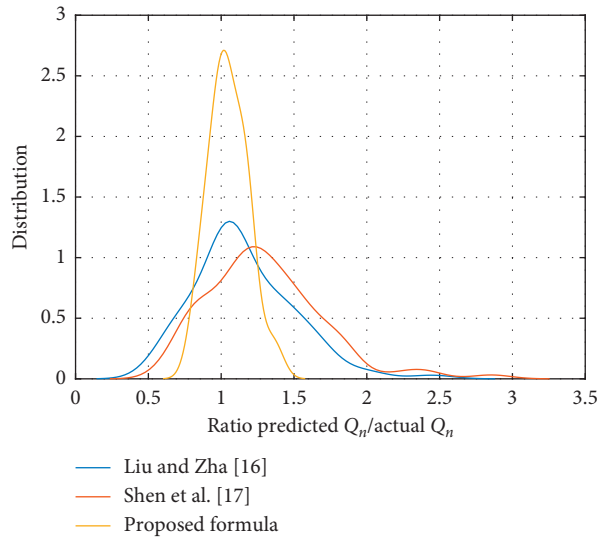
FIGURE 11: Distribution of ratio predicted Q_n/actual Q_n using different equations.

(a)   (b)
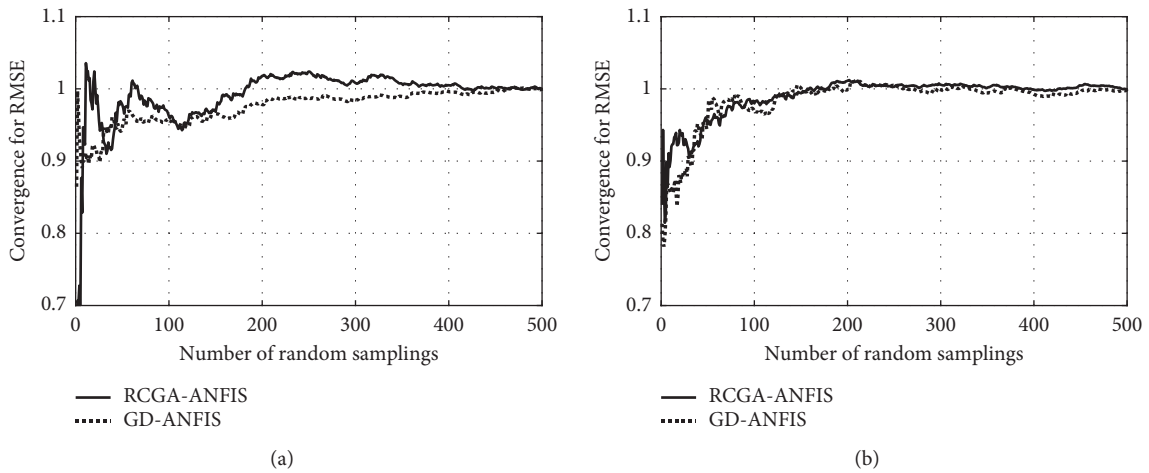
FIGURE 12: Continued.

(c)

(d)
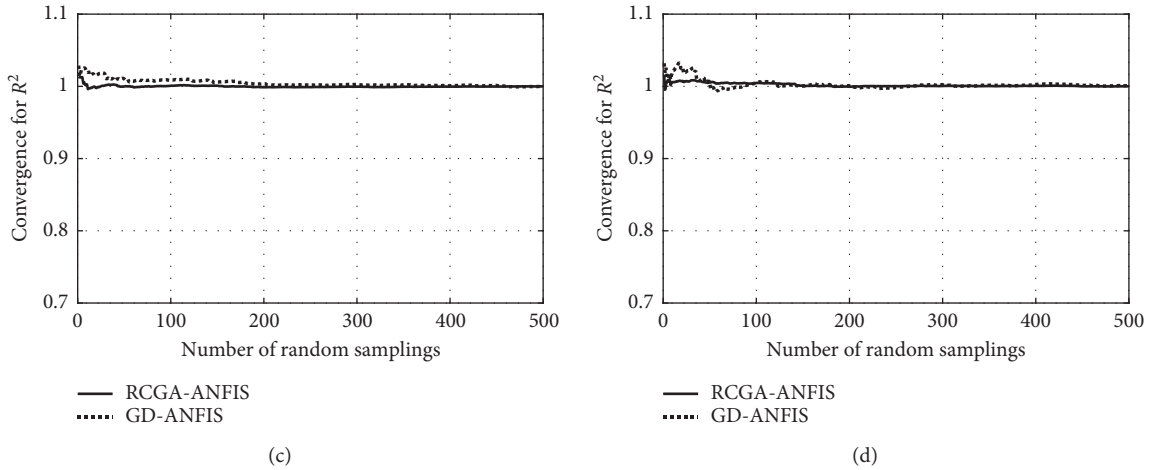
Figure 12: Monte Carlo convergence for training data: (a) RMSE, (b) R^2; for testing data: (c) RMSE, (d) R^2.

variation values. Finally, Figure 11 shows the probability density distribution of the three ratios.

It is seen in Table 7 (statistics of the three ratios) and Figure 11 that the prediction based on the proposed formula exhibits the highest agreement with the experimental data points or, in other words, the lowest error measurements (an average value of 1.05 compared to 1.16, 1.30 using Liu and Shen equations; a standard deviation value of 0.13 compared to 0.34, 0.40 using Liu and Shen equations; and a coefficient of variation of 12.55% compared to 29.27, 30.62 using Liu and Shen equations, respectively). It can be concluded that the prediction performance based on the proposed formula is superior to those available in the literature. Thus, with a simple form, the proposed formula can be used in practice. Moreover, if more experimental data are available in the future, the model will be improved (i.e., for a wider range of data).

4. Conclusions

The research presented in this article proposed a robust surrogate tool for the estimation of the ultimate load of elliptical CFST members under axial compression. Based on the developments and analyses, the following conclusions may be made:

(i) An experimental dataset was collected from the available literature for the development of the models including two groups of variables: geometric dimensions of cross section and mechanical properties of constituent materials (concrete and steel).

(ii) Two hybrid ML models, namely, the conventional GD-ANFIS and metaheuristic-based RCGA-ANFIS, were proposed to predict the ultimate load of the columns. The results showed that the RCGA-ANFIS model outperformed GD-ANFIS. In addition, the performance of the RCGA-ANFIS model was superior to two empirical equations in the literature.

(iii) The robustness of the proposed models was assessed by conducting Monte Carlo simulations taking into account the variability in the input space.

(iv) Sensitivity analysis showed that the steel pipe wall thickness and the short side length of the cross section were the most critical parameters affecting the bearing capacity of elliptical CFST columns (i.e., 22.264% and 21.344%, respectively).

(v) A Graphical User Interface was developed and provided freely for researchers/engineers/interested users. The results of the present work could simplify the design of elliptical CFST columns. The optimum values obtained in this study could allow quick and accurate determining of the bearing capacity of elliptical CFST columns for practical purposes.

However, it is worth noticing that, in this research, only elliptical CFST columns were considered. It is well-known that the cross section of columns has other forms; thus, the extension of the GUI to other cross sections would be the main perspective of the next study. In further research, a generic model should be developed for different types of cross section (i.e., circular, rectangular, square, hexagonal, etc.). Such a model can be highly beneficial for the research and practical purposes. Finally, in terms of practical application, a GUI based on Excel should be developed for wider applicability.

Appendix

Convergence of Monte Carlo simulations

In this section, the convergence of the ML models in the function of Monte Carlo runs is investigated (see Section 2.2.3). Figure 12 shows the convergence estimation in terms of RMSE and R^2, using the training and testing data, respectively. Regarding the convergence of R^2 for both training and testing part, low order of fluctuation was observed compared to RMSE. The statistical convergence analyses

showed that at least 500 Monte Carlo simulations were needed to obtain reliable results, particularly in terms of RMSE.

Data Availability

The Excel format data used to support the findings of this study may be made available upon request to Dr. Tien-Thinh Le, who can be contacted via thinh.letien@phenikaa-uni.edu.vn.

Conflicts of Interest

The author declares that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

References

[1] J. Yang, T. Sheehan, X. H. Dai, and D. Lam, "Experimental study of beam to concrete-filled elliptical steel tubular column connections," *Thin-Walled Structures*, vol. 95, pp. 16–23, 2015.

[2] F. McCann, "Concrete-filled elliptical section steel columns under concentric and eccentric loading," in *Proceedings of the 8th International Conference on Advances in Steel Structures, 2015*, Lisbon, Portugal, 2015.

[3] A. Espinos, L. Gardner, M. L. Romero, and A. Hospitaler, "Fire behaviour of concrete filled elliptical steel columns," *Thin-Walled Structures*, vol. 49, no. 2, pp. 239–255, 2011.

[4] S.-H. Lee, B. Uy, S.-H. Kim, Y.-H. Choi, and S.-M. Choi, "Behavior of high-strength circular concrete-filled steel tubular (CFST) column under eccentric loading," *Journal of Constructional Steel Research*, vol. 67, no. 1, pp. 1–13, 2011.

[5] T.-T. Le, "Surrogate neural network model for prediction of load-bearing capacity of CFSS members considering loading eccentricity," *Applied Sciences*, vol. 10, no. 10, p. 3452, 2020.

[6] H. Thanh Duong, H. Chi Phan, T.-T. Le, and N. Duc Bui, "Optimization design of rectangular concrete-filled steel tube short columns with balancing composite motion optimization and data-driven model," *Structures*, vol. 28, pp. 757–765, 2020.

[7] F. Ding, X. Ding, X. Liu, H. Wang, Z. Yu, and C. Fang, "Mechanical behavior of elliptical concrete-filled steel tubular stub columns under axial loading," *Steel and Composite Structures*, vol. 25, no. 3, pp. 375–388, 2017.

[8] H.-B. Ly, B. T. Pham, L. M. Le, T.-T. Le, V. M. Le, and P. G. Asteris, "Estimation of axial load-carrying capacity of concrete-filled steel tubes using surrogate models," *Neural Computing and Applications*, 2020, In press.

[9] L. Dennis and T. Nicola, "Structural design of concrete filled steel elliptical hollow sections," *Composite Construction in Steel and Concrete VI*, American Society of Civil Engineers, Reston, VA, USA, 2008.

[10] Y. Xu, J. Yao, and X. Sun, "Cold-formed elliptical concrete-filled steel tubular columns subjected to monotonic and cyclic axial compression," *Advances in Structural Engineering*, vol. 23, no. 7, p. 1383, 2019.

[11] T. M. Chan, L. Gardner, and K. H. Law, "Structural design of elliptical hollow sections: a review," *Proceedings of the Institution of Civil Engineers—Structures and Buildings*, vol. 163, no. 6, pp. 391–402, 2010.

[12] Q.-X. Ren, L.-H. Han, D. Lam, and W. Li, "Tests on elliptical concrete filled steel tubular (CFST) beams and columns," *Journal of Constructional Steel Research*, vol. 99, pp. 149–160, 2014.

[13] V. Q. Tran, H. L. Nguyen, V. D. Dao et al., "Effect of temperature on the chloride binding capacity of cementitious materials," *Magazine of Concrete Research*, pp. 1–39, 2019, In press.

[14] F. McCann and L. Gardner, "Numerical analysis and design of slender elliptical hollow sections in bending," *Thin-Walled Structures*, vol. 139, pp. 196–208, 2019.

[15] Y. Xu and J. Yao, "Axial bearing capacity of elliptical concrete filled steel tubular stub columns," *IOP Conference Series: Materials Science and Engineering*, vol. 220, Article ID 012002, 2017.

[16] X. C. Liu and X. X. Zha, "Study on behavior of elliptical concrete filled steel tube members I: stub and long columns under axial compression," *Progress in Steel Building Structures*, vol. 1, pp. 8–14, 2011.

[17] Q. H. Shen, J. F. Wang, W. Wang, and J. Wang, "Axial compressive behavior and bearing capacity calculation of ECFST columns based on numerical analysis," *Progress in Steel Building Structures*, vol. 6, pp. 68–78, 2015.

[18] Q. H. Nguyen, H.-B. Ly, T.-T. Le et al., "Parametric investigation of particle swarm optimization to improve the performance of the adaptive neuro-fuzzy inference system in determining the buckling capacity of circular opening steel beams," *Materials*, vol. 13, no. 10, p. 2210, 2020.

[19] H.-B. Ly, T.-T. Le, H.-L. T. Vu, V. Q. Tran, L. M. Le, and B. T. Pham, "Computational hybrid machine learning based prediction of shear capacity for steel fiber reinforced concrete beams," *Sustainability*, vol. 12, no. 7, p. 2709, 2020.

[20] H. Q. Nguyen, H.-B. Ly, V. Q. Tran, T.-A. Nguyen, T.-T. Le, and B. T. Pham, "Optimization of artificial intelligence system by evolutionary algorithm for prediction of axial capacity of rectangular concrete filled steel tubes under compression," *Materials*, vol. 13, no. 5, p. 1205, 2020.

[21] Q. H. Nguyen, H.-B. Ly, V. Q. Tran et al., "A novel hybrid model based on a feedforward neural network and one step secant algorithm for prediction of load-bearing capacity of rectangular concrete-filled steel tube columns," *Molecules*, vol. 25, no. 15, p. 3486, 2020.

[22] P. Sarir, J. Chen, P. G. Asteris, D. J. Armaghani, and M. M. Tahir, "Developing GEP tree-based, neuro-swarm, and whale optimization models for evaluation of bearing capacity of concrete-filled steel tube columns," *Engineering with Computers*, 2019, In press.

[23] M. Ahmadi, H. Naderpour, and A. Kheyroddin, "Utilization of artificial neural networks to prediction of the capacity of CCFT short columns subject to short term axial load," *Archives of Civil and Mechanical Engineering*, vol. 14, no. 3, pp. 510–517, 2014.

[24] M. Ahmadi, H. Naderpour, and A. Kheyroddin, "ANN model for predicting the compressive strength of circular steel-confined concrete," *International Journal of Civil Engineering*, vol. 15, no. 2, pp. 213–221, 2017.

[25] V.-L. Tran, D.-K. Thai, and S.-E. Kim, "Application of ANN in predicting ACC of SCFST column," *Composite Structures*, vol. 228, Article ID 111332, 2019.

[26] E. M. Güneyisi, A. Gültekin, and K. Mermerdaş, "Ultimate capacity prediction of axially loaded CFST short columns," *International Journal of Steel Structures*, vol. 16, no. 1, pp. 99–114, 2016.

[27] J. Moon, J. J. Kim, T.-H. Lee, and H.-E. Lee, "Prediction of axial load capacity of stub circular concrete-filled steel tube using fuzzy logic," *Journal of Constructional Steel Research*, vol. 101, pp. 184–191, 2014.

[28] K. Uenaka, "Experimental study on concrete filled elliptical/oval steel tubular stub columns under compression," *Thin-Walled Structures*, vol. 78, pp. 131–137, 2014.

[29] H. Yang, F. Liu, T.-m. Chan, and W. Wang, "Behaviours of concrete-filled cold-formed elliptical hollow section beam-columns with varying aspect ratios," *Thin-Walled Structures*, vol. 120, pp. 9–28, 2017.

[30] F. Liu, Y. Wang, and T.-m. Chan, "Behaviour of concrete-filled cold-formed elliptical hollow sections with varying aspect ratios," *Thin-Walled Structures*, vol. 110, pp. 47–61, 2017.

[31] X. H. Dai, D. Lam, N. Jamaluddin, and J. Ye, "Numerical analysis of slender elliptical concrete filled columns under axial compression," *Thin-Walled Structures*, vol. 77, pp. 26–35, 2014.

[32] N. Jamaluddin, D. Lam, X. H. Dai, and J. Ye, "An experimental study on elliptical concrete filled columns under axial compression," *Journal of Constructional Steel Research*, vol. 87, pp. 6–16, 2013.

[33] H. Yang, D. Lam, and L. Gardner, "Testing and analysis of concrete-filled elliptical hollow sections," *Engineering Structures*, vol. 30, no. 12, pp. 3771–3781, 2008.

[34] F. McCann, L. Gardner, and W. Qiu, "Experimental study of slender concrete-filled elliptical hollow section beam-columns," *Journal of Constructional Steel Research*, vol. 113, pp. 185–194, 2015.

[35] X. L. Zhao and J. A. Packer, "Tests and design of concrete-filled elliptical hollow section stub columns," *Thin-Walled Structures*, vol. 47, no. 6-7, pp. 617–628, 2009.

[36] I. T. Jolliffe, *Principal Component Analysis*, Springer-Verlag, New York, NY, USA, 2nd edition, 2002.

[37] J.-S. R. Jang, "ANFIS: adaptive-network-based fuzzy inference system," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 23, no. 3, p. 665, 1993.

[38] J.-S. R. Jang, C.-T. Sun, and E. Mizutani, *Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence*, Prentice-Hall, Upper Saddle River, NJ, USA, 1997.

[39] M. Bilgehan, "Comparison of ANFIS and NN models—with a study in critical buckling load estimation," *Applied Soft Computing*, vol. 11, no. 4, pp. 3779–3791, 2011.

[40] H.-L. Nguyen, B. T. Pham, L. H. Son et al., "Adaptive network based fuzzy inference system with meta-heuristic optimizations for international roughness index prediction," *Applied Sciences*, vol. 9, no. 21, p. 4715, 2019.

[41] H.-L. Nguyen, T.-H. Le, C.-T. Pham et al., "Development of hybrid artificial intelligence approaches and a support vector machine algorithm for predicting the marshall parameters of stone matrix asphalt," *Applied Sciences*, vol. 9, no. 15, p. 3172, 2019.

[42] H.-B. Ly, L. M. Le, L. V. Phi et al., "Development of an AI model to measure traffic air pollution from multisensor and weather data," *Sensors*, vol. 19, no. 22, p. 4941, 2019.

[43] H.-B. Ly, L. M. Le, H. T. Duong et al., "Hybrid artificial intelligence approaches for predicting critical buckling load of structural members under compression considering the influence of initial geometric imperfections," *Applied Sciences*, vol. 9, no. 11, p. 2258, 2019.

[44] D. Dao, H.-B. Ly, S. Trinh, T.-T. Le, and B. Pham, "Artificial intelligence approaches for prediction of compressive strength of geopolymer concrete," *Materials*, vol. 12, no. 6, p. 983, 2019.

[45] J. Holland, *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor, MI, USA, 1975.

[46] M. Mitchell, *An Introduction to Genentic Algorithms*, MIT Press, Cambridge, MA, USA, 1998.

[47] D. Whitley, "A genetic algorithm tutorial," *Statistics and Computing*, vol. 4, pp. 65–85, 1994.

[48] W. Wan and J. B. Birch, "An improved hybrid genetic algorithm with a new local search procedure," *Journal of Applied Mathematics*, vol. 2013, Article ID 103591, 10 pages, 2013.

[49] H.-F. Wang and K.-Y. Wu, "Hybrid genetic algorithm for optimization problems with permutation property," *Computers & Operations Research*, vol. 31, no. 14, pp. 2453–2471, 2004.

[50] H.-j. Kim and K.-s. Shin, "A hybrid approach based on neural networks and genetic algorithms for detecting temporal patterns in stock markets," *Applied Soft Computing*, vol. 7, no. 2, pp. 569–576, 2007.

[51] L. M. Le, H.-B. Ly, B. T. Pham et al., "Hybrid artificial intelligence approaches for predicting buckling damage of steel columns under axial compression," *Materials*, vol. 12, no. 10, p. 1670, 2019.

[52] X. Yan, H. Liu, Z. Zhu, and Q. Wu, "Hybrid genetic algorithm for engineering design problems," *Cluster Computing*, vol. 20, no. 1, pp. 263–275, 2017.

[53] C. K. H. Lee, "A review of applications of genetic algorithms in operations management," *Engineering Applications of Artificial Intelligence*, vol. 76, pp. 1–12, 2018.

[54] B. Staber, J. Guilleminot, C. Soize, J. Michopoulos, and A. Iliopoulos, "Stochastic modeling and identification of a hyperelastic constitutive model for laminated composites," *Computer Methods in Applied Mechanics and Engineering*, vol. 347, pp. 425–444, 2019.

[55] T. T. Le, J. Guilleminot, and C. Soize, "Stochastic continuum modeling of random interphases from atomistic simulations. Application to a polymer nanocomposite," *Computer Methods in Applied Mechanics and Engineering*, vol. 303, pp. 430–449, 2016.

[56] J. Guilleminot, T. T. Le, and C. Soize, "Stochastic framework for modeling the linear apparent behavior of complex materials: application to random porous materials with interphases," *Acta Mechanica Sinica*, vol. 29, no. 6, pp. 773–782, 2013.

[57] H.-B. Ly, E. Monteiro, T.-T. Le et al., "Prediction and sensitivity analysis of bubble dissolution time in 3D selective laser sintering using ensemble decision trees," *Materials*, vol. 12, no. 9, p. 1544, 2019.

[58] J. Guilleminot and C. Soize, "Stochastic model and generator for random fields with symmetry properties: application to the mesoscopic modeling of elastic random media," *Multiscale Modeling & Simulation*, vol. 11, no. 3, pp. 840–870, 2013.

[59] D. A. Hun, J. Guilleminot, J. Yvonnet, and M. Bornert, "Stochastic multiscale modeling of crack propagation in random heterogeneous media," *International Journal for*

Numerical Methods in Engineering, vol. 119, no. 13, pp. 1325–1344, 2019.

[60] S. Chu and J. Guilleminot, "Stochastic multiscale modeling with random fields of material properties defined on non-convex domains," *Mechanics Research Communications*, vol. 97, pp. 39–45, 2019.

[61] T.-T. Le, *Modélisation stochastique, en mécanique des milieux continus, de l'interphase inclusion-matrice à partir de simulations en dynamique moléculaire*, Ph.D. thesis, University of Paris-Est Marne-la-Vallée, Paris, France, 2015.

[62] D. V. Dao, H. Adeli, H.-B. Ly et al., "A sensitivity and robustness analysis of GPR and ANN for high-performance concrete compressive strength prediction using a Monte Carlo simulation," *Sustainability*, vol. 12, no. 3, p. 830, 2020.

[63] T.-T. Le, "Probabilistic investigation of the effect of stochastic imperfect interfaces in nanocomposites," *Mechanics of Materials*, vol. 151, Article ID 103608, 2020.

[64] B. Staber and J. Guilleminot, "Stochastic modeling of a class of stored energy functions for incompressible hyperelastic materials with uncertainties," *Comptes Rendus Mécanique*, vol. 343, no. 9, pp. 503–514, 2015.

[65] J. Guilleminot and J. E. Dolbow, "Data-driven enhancement of fracture paths in random composites," *Mechanics Research Communications*, vol. 103, Article ID 103443, 2020.

[66] D. V. Dao, H.-B. Ly, H.-L. T. Vu, T.-T. Le, and B. T. Pham, "Investigation and optimization of the C-ANN structure in predicting the compressive strength of foamed concrete," *Materials*, vol. 13, no. 5, p. 1072, 2020.

[67] B. T. Pham, T. Nguyen-Thoi, H.-B. Ly et al., "Extreme learning machine based prediction of soil shear strength: a sensitivity analysis using Monte Carlo simulations and feature backward elimination," *Sustainability*, vol. 12, no. 6, p. 2339, 2020.

[68] H.-B. Ly, T.-T. Le, L. M. Le et al., "Development of hybrid machine learning models for predicting the critical buckling load of I-shaped cellular beams," *Applied Sciences*, vol. 9, no. 24, p. 5458, 2019.

[69] T.-T. Le, "Probabilistic modeling of surface effects in nano-reinforced materials," *Computational Materials Science*, vol. 186, Article ID 109987, 2021.

[70] T.-T. Le, "Prediction of tensile strength of polymer carbon nanotube composites using practical machine learning method," *Journal of Composite Materials*, 2020, In press.

[71] İ1 Güler and E. D. Übeyli, "Adaptive neuro-fuzzy inference system for classification of EEG signals using wavelet coefficients," *Journal of Neuroscience Methods*, vol. 148, no. 2, pp. 113–121, 2005.

[72] A. F. Mashaly and A. A. Alazba, "ANFIS modeling and sensitivity analysis for estimating solar still productivity using measured operational and meteorological parameters," *Water Supply*, vol. 18, no. 4, pp. 1437–1448, 2018.

[73] MATLAB, *The MathWorks*, MATLAB, Natick, MA, USA, 2018.

[74] I. M. Nikbin, S. Rahimi, and H. Allahyari, "A new empirical formula for prediction of fracture energy of concrete based on the artificial neural network," *Engineering Fracture Mechanics*, vol. 186, pp. 466–482, 2017.

Research Article

Prediction of the Bending Strength of Boltless Steel Connections in Storage Pallet Racks: An Integrated Experimental-FEM-SVM Methodology

Zhi-Jun Lyu [1,2] **PeiCai Zhao** [1,2] **Qi Lu,** [3] **Qian Xiang,** [1,2] **and HongLiang Li** [2]

[1]*College of Mechanical Engineering, Donghua University, Shanghai 201620, China*
[2]*Shanghai Engineering Research Center of Storage & Logistics Equipment, Shanghai 201611, China*
[3]*SAIC General Motors, Shanghai 201206, China*

Correspondence should be addressed to Zhi-Jun Lyu; lvzj@dhu.edu.cn and PeiCai Zhao; 852692515@qq.com

Due to many differences in the material, geometry, and assembly method of the commercially available beam-end-connectors in steel storage pallet racks (SPR), no common numerical model has been universally accepted to accurately predict the M–θ behavior of complex semirigid connections so far. Despite the fact that the finite element method (FEM) and physical experiment have been used to obtain the mechanical performance of beam-to-column connections (BCCs), those methods have the disadvantages of high computational complexity and test cost. Taking, for example, the boltless steel connections, this paper proposes a data-driven simulation model (DDSM) that combines the experimental test, FEM, and support vector machine (SVM) techniques to determine the bending strength of BCCs by means of data mining from the engineering database. First, a three-dimensional (3D) finite element (FE) model was generated and calibrated against the experimental results. Subsequently, the validated FE model was further extended to perform parametric analysis and enrich the engineering case base of structural characterization of BCCs. Based on the M–θ curve of the FE simulation, support vector machines (SVMs) were trained to predict the flexural rigidity of beam-to-column joints. The predictive power of the SVM algorithms is estimated by comparison with traditional ANN models via the root mean square error (RMSE), the mean absolute percentage error (MAPE), and the correlation coefficient R. The results obtained indicate that the SVM algorithms slightly outperform the ANN algorithms, although both of them are in good agreement with FEM and physical test. From the point of view of engineering application, DDM is able to provide much more effective help for structural engineers to make rapid decisions on steel members design.

1. Introduction

With the rapid advancement of e-commerce, automated storage and retrieval systems (AS/RS) have been so widely applied in China that high-rise steel storage pallet racks (SPR) have exhibited an explosive growth in production and logistics system (Figure 1). Acting as one of the most important infrastructures for AS/RS, structural design for SPR needs the elaborate decision-making between structural systems and a variety of cold-formed steel members in such a way that the stability and safety behave as intended by the designer and satisfies the constraints imposed by capital investment, environment, and so on. Opposite to traditional civil engineering structures, the material of steel members in storage pallet racking is thin and lightweight while the racking system itself can usually carry live load many times larger than the dead load with an extraordinary height.

Of all the members in the SPR, the beam-to-column connections (BCCs) constitute the most critical part of the assembly which largely determines the overall stability of SPR in the down-aisle direction [1]. The details of the boltless BCCs with the three rivets mostly used in industrial racking system are shown as an example in Figure 2. Due to the great variety of connector types and connected members, a generalized analytical evaluation of the connection mechanical properties still appears to be very difficult [2]. One

FIGURE 1: High-rise steel pallet racks under construction in China.

(a)   (b)   (c)
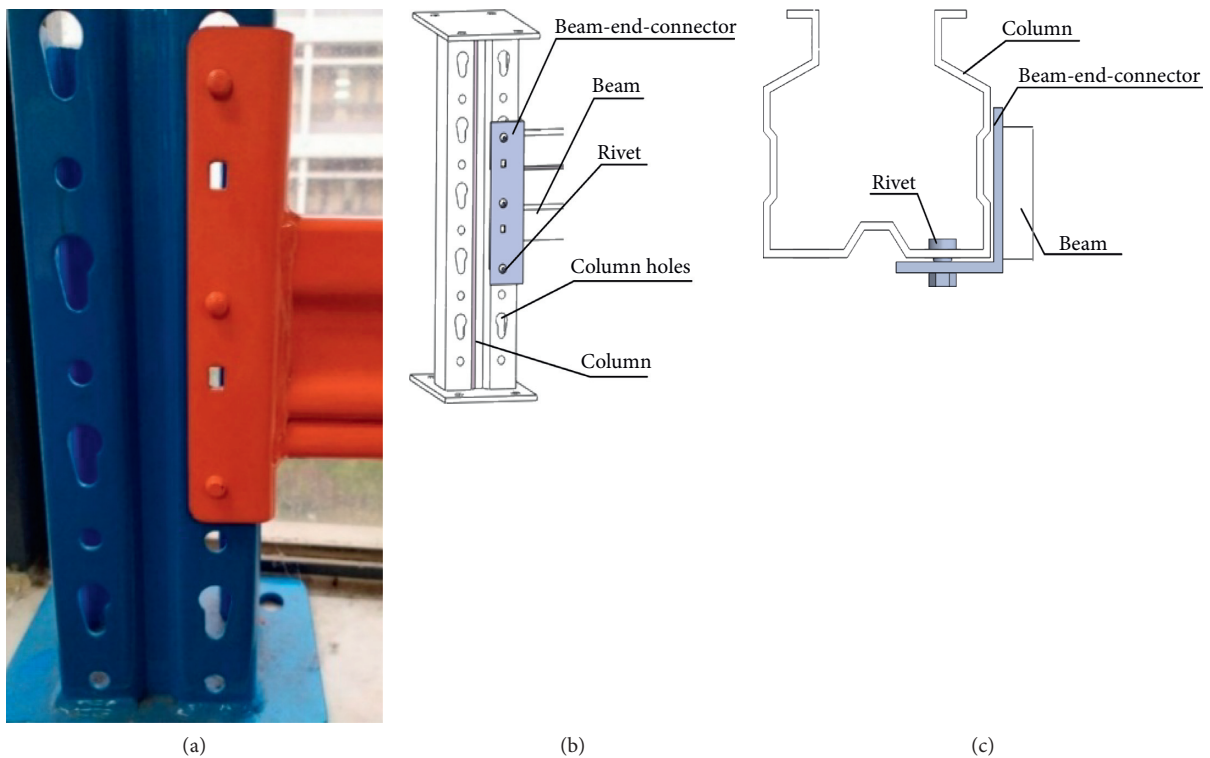
FIGURE 2: Details of SPR BCC. (a) Physical map. (b) Top view. (c) Isometric drawing.

of main reasons for this is that the typical boltless steel connections are essentially called "semirigid" or "partial-strength" structure representing a strong nonlinear behavior [3]. Therefore, the most recent design codes, such as those of the EN 15512 [4], RMI [5], and AS4084 [6], recommend physical experiment method of the testing results to assess the moment-rotation (M-θ) behavior of any SPR BCC. Numerous studies in the last few years are available on the experimental testing of SPR BCCs [7–11]. Apparently, these investigations, dependent on experimental results, are relatively accurate and reliable but their arrangements are too expensive, and operations are too complicated to be utilized in industrial production on a large scale. On the other hand, the possibility of random or systematic errors in the experimental investigations and the diversity of beam-end-connectors also directed researchers towards the finite element (FE) modeling

of connections [2, 12]. FE modeling, which was made by using different analysis software, has proven itself to be a powerful tool to gain a more predictable performance of the connections and the effects of various parameters on the overall performance of SPR. Furthermore, a suitable solution was proposed to derive a particular uniform M-θ relationship for each connection type on the basis of BCCs geometric parameters through experimental investigations and finite element (FE) modeling [13]. From analytic point of view, Zhao et al. and Gusella et al. proposed a mechanical model based on the component method to predict the initial rotational stiffness of beam-to-upright connections [14, 15]. Those analytical models are based on physical knowledge of stability mechanics, which not only are very appealing but also give a thorough insight into the deformation mechanism of multiple components. However, to cope with the inherent complexity of structural mechanics, some assumptions have to be introduced into these models, among which, the assembly relationship of the beam-to-column joints has not specially been addressed in previous studies. This may ultimately affect the prediction accuracy of analytical models and then result in unexpected deviation from physical tests. The increasing demands for cold-formed thin-walled steel in modern industry need to explore more reliable methods of accurate prediction of the behavior of storage racks, which have a wide range of adaptability and operational convenience in engineering design. The proliferation of industrial "big data" has created many exciting opportunities for those working in various fields such as science, engineering, and business. The machine learning (ML) and data mining (DM) from industrial big data have been rapidly developed as new disciplines of computer science and engineering application [16, 17]. It has been gradually realized that those data from engineering experiments and analysis not only can be used for the engineering practice, but also have the potential to provide insight and knowledge for the designer to improve the construction quality itself. The data-driven approaches focus on analysis and discovery of the potential pattern of design process and can realize precise prediction of complex engineering problems, usually including some metaheuristic optimization algorithms such as the genetic algorithm and particle swarm optimization, artificial neural network (ANN), support vector machines, and Bayesian models [18]. Within the constructional steel fields, the advantage of an ANN was used to propose an intelligent finite element for viscoelastic material behavior in [19]; Shah et al. [2] also proposed a hybrid intelligence model based on linear genetic programming (LGP), artificial neural networks (ANNs), and adaptive neuro-fuzzy inference system (ANFIS) to predict the moment-rotation (M-θ) behavior of boltless steel connections [20].

In recent years, a variety of machine learning methods have been applied on a large scale in the modern industrial and civil engineering field. Among them, convolutional neural network (CNN) is one of the representatives of deep learning algorithm, which is suitable for multipixel and audio processing. Cha et al. [21] and Wang and Cha [22] used these novel deep learning methods for damage detection in structural-health monitoring for civil structures.

In addition, Santos et al. [23] compared four kernel-based algorithms for damage detection under varying operational and environmental conditions, namely, based on one-class support vector machine, support vector data description, kernel principal component analysis, and greedy kernel principal component analysis. Langone et al. [24] came up with a technique called adaptive kernel spectral clustering (AKSC) which unifies the data normalization and damage detection steps. Inspired by the idea of unsupervised feature learning that uses artificial intelligence techniques to learn features from raw data, a two-stage learning method is proposed, with Moving Kernel Principal Component Analysis (MKPCA) and Nyström methods, by Ghiasi and Ghasemi [25] for intelligent health monitoring of civil engineering structures. Besides, support vector machines (SVMs) are also receiving increasing attention in different application domains for which artificial neural networks (ANNs) have had a prominent role, due to their many attractive features and promising empirical performance. This systematic approach, motivated by statistical learning theory, led to a class of algorithms characterized by the use of kernels, the absence of local examples, the sparseness of the solution, and the capacity control obtained by acting on the margin. Unlike traditional ANN models, SVM models are based on the principle of structure risk minimization (SRM), which equips the latter with greater potential to generalize. Since the foundation of the SVMs paradigm was laid down by Vapnik in mid-1998 [26], applications in many engineering fields have emerged, such as architecture [27], communication system [28], geology [29], and even financial management [30]. However, reports about which SVMs are used for predicting the M-θ behavior of SPR BCCs have not been seen so far.

Taking the riveted BCCs as our research object, we present a novel data-driven model, using an integrated experimental-FEM-SVM methodology to overcome many difficulties associated with the mechanical performance of semirigid beam-to-upright joint modeling, which is the main contribution of this paper. The objective of data-driven based predictive models is the development of enabling tools for designers to make rapid and effective decision when big datasets are available on prediction and reasonable number of predictors. Compared with existing references, the obvious distinctions of our work lie in the fact that the finite element simulation data based on physical test are utilized to train SVM model and predict the bending strength of the complex boltless steel connections with data mining method. The results have undergone comparative analysis with those of the traditional FEM and ANN. The preliminary investigation demonstrates that the data-driven models have a reasonably good accuracy in most of the cases and are more suitable for the nonlinear mechanical behaviors. The outline of the remaining content of this paper is as follows. Section2 briefly describes data-driven model framework and integrated methodology. Based on the data from physical performance tests of BCCs, Section 3 develops a finite element model to exactly simulate the flexural behavior under monotonic loads. Section 4 introduces the SVM regression algorithm and data mining process. The results and discussion of the case study

are presented in Section 5. Finally, the conclusions and future work are summarized in Section 6.

2. Data-Driven Based Methodology Framework

Generally speaking, the current models available to solve the mechanical performance problem of the SPR BCCs can be mainly categorized into two types: physical experiment based analytical models and FE based numerical model. However, the use of the two techniques for analyzing this wide range of beam-to-upright assembly in massive engineering practices could be inappropriate due to the great amount of time and economic resources required. The methodology proposed in our paper is based on a hybrid approach of experimental, numerical (finite element method), and machine learning (support vector machines) techniques, which allows the obtainment of computational efficient results for various design solutions to make rapid and accurate evaluation. As shown in Figure 3, the data-driven modeling framework includes three stages, and the general task in each stage is described as follows.

Stage I: Data acquisition

The task of this stage is to collect and transform the data from the beam-to-column physical experiment and finite element simulation into engineering database. Because the physical experiments are so costly that volume of real dataset is relatively limited, the finite element simulation is employed to expand engineering data as machine learning required. On the basis of the test data, the finite element model in the commercial software ANSYS is repeatedly calibrated and validated in order to exactly simulate the blending process of the cantilever beam experiment; then, using the so-called virtual testing method, the different rotational stiffness from finite element simulation for the existing joint solutions is obtained instead of the real physical test. Finally, the substantial data such as the geometric features, assembly relationship, and corresponding mechanical behavior on the diverse BCC joints are stored in engineering analysis database.

Stage II: Machine learning

This stage is the core module of data-driven modeling which can cover the full machine learning pipeline from data processing to result evaluation. In most cases, those modeling data from the engineering database fall within different ranges. It is highly essential to pre-process the input data before applying them to the machine learning models, so as not to affect the obtained results. On the other hand, these raw data and engineered features probably have a large number of independent or redundant variables, which often make models more complex and incomprehensible. There are two main dimensionality reduction methods for data: one is to extract the main features of the data by destroying the original structure of the data. The other is to conduct correlation analysis on the data and select the attributes of the data according to certain rules to
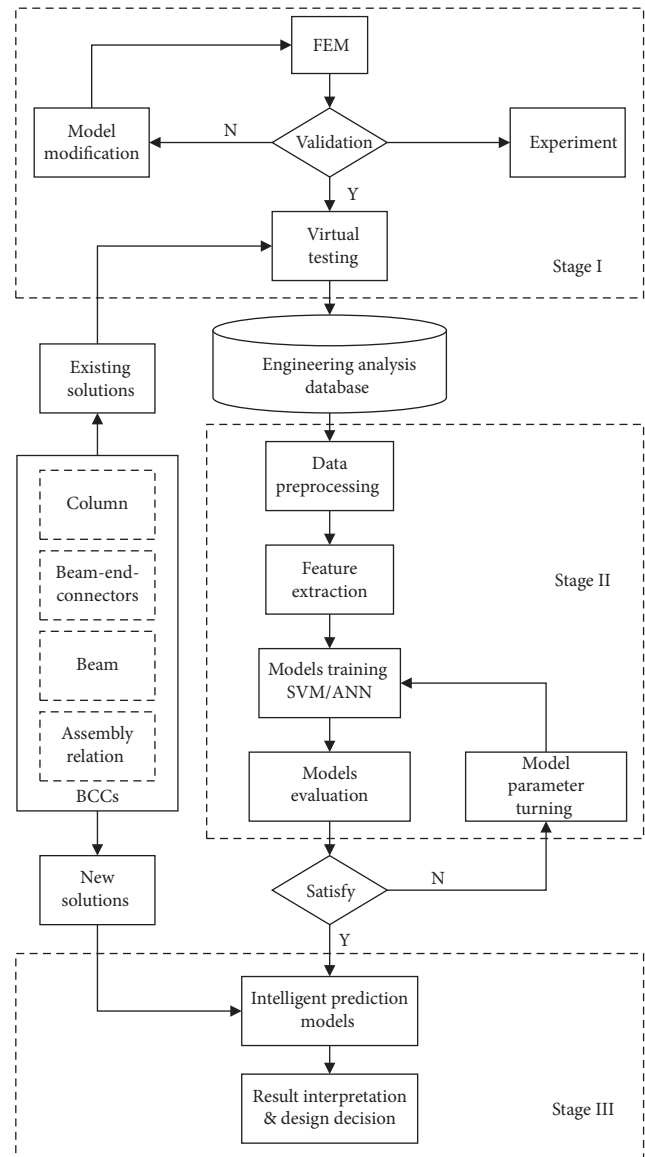
FIGURE 3: Methodology framework.

achieve the purpose of dimensionality reduction. Kernel methods belonging to the first type, such as the kernel principal component analysis (KPCA), have the ability to find nonlinear patterns from the data while keeping the computational elegance of matrix algebra, but they often take up a lot of memory and the calculation is more complicated [31]. Here, the correlation coefficient after Pearson R falls into the second type as an easier feature extraction method is used to reduce the data dimensionality and improve the generalization performance of a predictive model. In the model training, the normalized dataset is randomly divided into separate train and test sets; on the basis of those data, the control parameters of the SVM model are continuously adjusted and optimized through iterative loop mode until the predictive accuracy satisfies the need of engineering practice as a whole.

Stage III: Design decision

Once the predictive model is trained well, new design solutions for the BCC joint are input into it one by one, and their mechanical performance can be quickly obtained so that structural engineers can make more reasonable decisions.

Unlike the existing programs and methods [2, 14, 32, 33], the merits of data-driven model framework lie in the following:

(i) SVM/ANN is a self-adaptive and data-driven method in nature, so there is no need to make some rigorous assumptions about the statistical distribution on real engineering data.

(ii) SVMs are good at handling data with much more features than samples, which makes it more accurate for modeling complex data patterns, as opposed to traditional modeling approaches based on a large amount of test data.

(iii) Along with the growth of engineering data, the proposed framework is very expandable and has the capability of improving prediction accuracy by system self-learning.

(iv) Robust reasoning machine in the intelligent prediction model is utilized to optimize design parameters on the SPR BCCs as predictive model calculating without consideration of the potential rule collision from explicit design knowledge.

3. Experiments Based Finite Element Simulation

The flexural tests on the different BCCs are the foundation of analyzing the mechanical performance of boltless connections. On the basis of the physical test, a refined finite element model on SPR BCCs was built to simulate the cantilever test process as accurate as possible in this section.

3.1. Experimental Program.
The cantilever testing method is considered to be an efficient method to predict the strength characteristics of SPR BCCs [34], which can give a precise experimental evaluation of the flexural behavior of locally manufactured beam-to-column connections when subjected to increasing static hogging loading. In this method, both ends of the column are kept rigidly fixed. The end of the pallet beam attached to the end connector is inserted in the perforations at the center of the column, and the other end is left in cantilever. A lateral restraint is provided to prevent the twisting of the beam end, and the beam is left free to move in the loading direction. Loading should be applied 610 mm from the face of the column. The consequent displacement in the line of action of the applied load and/or the rotation near the connector is observed. The rotation is measured by either transducers or inclinometers.

3.1.1. Material Properties.
The material properties of the column, beam, and beam-end-connector were obtained through the tensile coupon test and are given in Table 1. The

test was conducted according to EN 15512 [4]. The standard specifies that the testing specimens for tensile test should be cut from the direction of rolling on samples of raw material coil, in accordance with EN 10002-1 [35].

3.1.2. Specimen Details.
The specimens were distinguished by three different types of columns, four different beam-end-connector thicknesses (Table 2, Figure 4), three different beam positions (Figure 5), three different clearances between the connectors and the column webs (Figure 5), and other characteristics (see Section 5). All the dimensions of the specimens are the measured values.

3.1.3. Testing Arrangement.
Six groups of beam-to-columns chosen in pallet racking have been tested and analyzed from the existing experimental data which were collected from the Shanghai Jingxing Logistics Equipment Engineering Co., Ltd, China. The experiment setup and supporting systems are shown in Figure 6.

At the beginning of the test, an initial load F of 10% of the expected failure load was preloaded at 400 mm from the beam flange surface to the column. The purpose is to make the rivets on the beam-end-connector fully contact the column grooves, then fix the components, and then unload. The measuring instrument was reset, and then the force F was gradually increased to the maximum load value until the BCCs failed. During the test, load F was measured by a load cell, and the vertical components of the displacements d_1 and d_2 at the loaded section were directly monitored by the linear variable displacement transducer (LVDT) of the testing machine. LVDTs and wire-actuated encoders were connected to a computer-aided data recording system and load cells.

3.1.4. The Experimental Results and Moment-Rotation Response.
The stiffness of beam-to-column is obtained by moment-rotation (M-θ) curve. The rotation may be measured by displacement transducers bearing onto a plate tack-welded to the beam close to the connector, but with enough clearance to allow for connector distortion. The moment M and rotation θ were calculated by the following equations:

$$M = b \cdot F, \tag{1}$$

$$\theta = \frac{d_1 - d_2}{k}, \tag{2}$$

where F is the loading; b is the distance between the loading jack and the surface of the column, which is 400 mm; d_1, d_2 are the displacements; and k is the distance between d_1 and d_2.

According to the code EN 15512 [4], the yield stress and thickness of the materials of the beam, upright and connector in Table 2 are used to calibrate the observed value of M and θ of the test. The acquisition of stiffness requires an over coordinate origin line at the M-θ polynomial fitting curve (Figure 7), which, with the line of design moments $M_{t,\text{Rd}}$, divides the test curve into two equal parts (A_1 and A_2). Apparently, the bending strength of beam-to-column is determined by the slope of the line.

TABLE 1: Material properties of the specimens.

Structural elements	Young's modulus E (GPa)	Poisson's ratio ν	Yield strength f_y (N/mm^2)	Ultimate strength f_u (N/mm^2)
Column			376	562
Beam	210	0.3	348	496
Beam-end-connector			359	528

TABLE 2: Specimen from the columns, beams, and connectors.

Structural elements	Web length/beam width	Flange width/beam height	Mean thickness $t_{t,m}$ (mm)	Mean yield strength $f_{y,m}$ (N/mm^2)
Column	94	77	2.023	339.8
	100	100	2.055	375.8
Beam	48	100	1.51	348
Connector			4.030	358.5

FIGURE 4: Cross section of the connection components. (a) Column. (b) Beam. (c) Beam-end-connector.
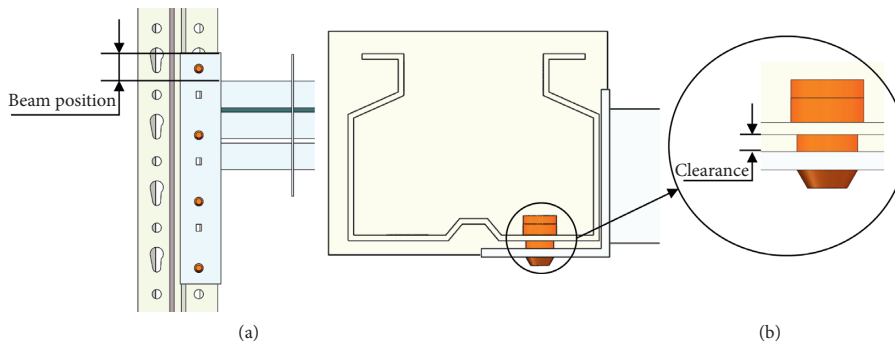
FIGURE 5: Assembly parameters. (a) Beam position of BCC. (b) Clearance of BCC.

3.2. Finite Element Simulation. The goal of the finite element (FE) model of the BCCs in our work is the flexibility to represent the complex cross-sectional geometry of the component and the ability to assign semirigid behavior to the seam and take into account the effects of local deformation, as well as flexural buckling of the ultimate bearing capacity of the structure. With reference to the finite element (FE) modeling of storage rack system [20], the commercial finite element software ANSYS R17 is used for the model development and analysis presented in this paper.

3.2.1. Establishment of Geometric Model. Firstly, the 3D models are established based on the real value of the tested samples using the SolidWorks software. It is noticed that the cross section and the hole setting are not simplified to ensure the accuracy of the finite element model, but the chamfer and fillet have been simplified accordingly. The beam and the measuring plate on the beam are welded in the actual test, so

the beam and the measuring plate are integrally modeled during the three-dimensional modeling, as shown in Figure 8. Different finite element types in the ANSYS software package are used in the modeling of beams, columns, connectors, and rivets. Among them, SHELL163 (4-node 3D elastic shell) was used to model beams, columns, and connectors. SOLID45 (8-node 3D structural solid) was used to model rivets and load plates. The spar elements carry only axial forces, and any shear on the interface between the connector and the column flange will be transferred through the friction allowed by the contact elements. Specific element characteristics are shown in Table 3.

3.2.2. Material Properties and Mesh. The materials on the tested samples were set as nonlinear steel for subsequent buckling analysis by the ANSYS Workbench with Structural Steel NL. The material properties of all the three components listed in Table 1 were imported to the FE model (FEM).
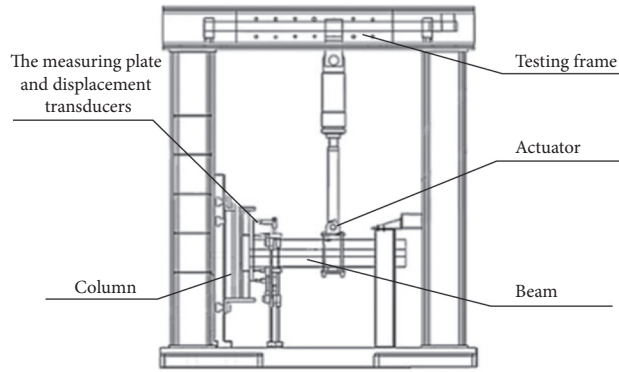
FIGURE 6: Schematic diagram of test setup.
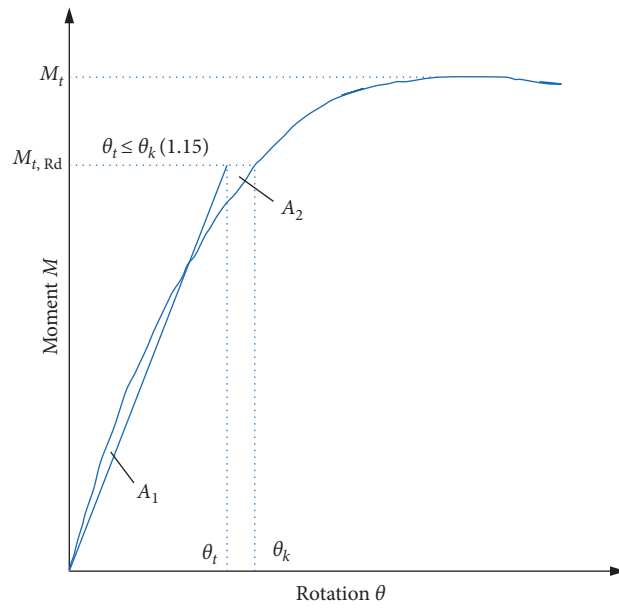
FIGURE 7: M-θ polynomial fitting curve for the typical riveted BCC.

(a)

(b)
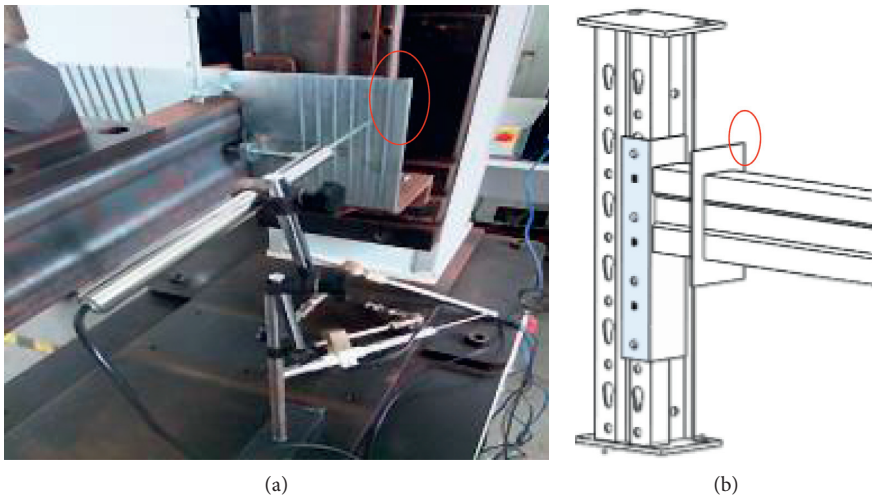
FIGURE 8: Measuring plate.

TABLE 3: Element characteristics of SHELL163 and SOLID45.

Name	SHELL163	SOLID45
Position	Column, beam, connector	Rivet
Character	Elastic shell element	3D solid structural unit
Node number	4	8
Nodal freedom	UX, UY, UZ and ROTX, ROTY, ROTZ	UX, UY, UZ

Meshing is the basis of finite element analysis. Reasonable meshing can reduce the use of computer memory, and the results are more accurate. Firstly, the 3D assembly model of beam-column joint is sliced in the geometry module, and the column, connector beam weldment, and rivet are manually split into three independent parts. Then the independent parts are from new part processed, and the model is divided into 20-node full hexahedron units. As the main structures for transmitting torque, the open hole parts of the upright posts, connectors, and rivet structures are more prone to deformation, so the grids of these parts are more finely divided. Numerical trials using different mesh sizes have shown that mesh size of 10 mm is more appropriate to achieve accurate results. After division, Figure 9 shows the meshing of the entire connection.

3.2.3. Surface Interaction. The beam-column joints are semirigid assemblies with welded and riveted components and different contact surfaces between members. The accuracy of simulation results will be affected by surface-to-surface contacts directly. Therefore, different contact settings are made for the contact surfaces that interfere when the beam-column joints are stressed. As shown in Figure 10, in the assembly of beam-column joints, except for the welded beams and mounting plates, the mounting plates and rivets with interference fit are not required to be set as a whole by default. The contact surface of each rivet matching with the column hole is shown in Figure 10(a). The contact surface on which the hanging plate and the column may deform under stress is shown in Figure 10(b). The contact between the rivet and the column hole is set to friction so that the connection between the beam and the column is loose and meets the semirigid property of the BCCs. Frictional parameters were set at 0.3 by referring to GB 50017-2003 steel structure design specification. During the simulation process, the contact surface between the column and the connector may be deformed without interfering with each other; in order to ensure the simulation accuracy, the contact surface parameters need to be set to the "adjust to touch" option (contact only).

3.2.4. Loading and Boundary Conditions. In the experimental setup, the bolt-type fixed connection was used to completely restrain the six degrees of freedom of the column. Further, the fixed frame was used to limit the displacement of the beam perpendicular to the column. Therefore, similar boundary conditions were applied to the end of the column and the beam. For the test setup, a force loading was applied to the beam at a distance of 400 mm from the contact surface of the column and the connector. Similar loading protocol

was adopted for the FE analysis. The load was applied to the top of the beam which causes compression in the top of the beam-end-connector and tension at the bottom. All node displacements of the bottom plates on column and side plates on beam in the negative Y axis have been set to zero (Figure 11). Two probe points were placed on the top flanges of the beams on either side to observe the deflection in the beams. The stiffness of FEM simulation could be obtained by M-θ curve as shown in Figure 7.

3.2.5. Validation of the FE Model. The simulation results of beam-column joints when the load F reaches 1000 N are shown in Figure 12. Referring to the method of obtaining stiffness value of beam-column joint in mechanical performance test, the four measuring points are, respectively, placed at four corners of beam measuring plate, corresponding to displacement sensors in mechanical performance test. Among them, the average value of the results of the two measuring points A and B is taken as the value of d_1 in formula (2), the average value of the results of the two measuring points C and point D is taken as the value of d_2 in formula (2), the M and θ values under the corresponding loads are obtained according to the calculation method in Section 3.1, and the M-θ curve of the finite element model is obtained by curve-fitting finally.

A comparison of the M–θ graphs plotted for the experimental and finite element studies is provided in Figures 13(a)–13(d). Four specimens with varying column thickness values and column cross-sectional areas were compared to illustrate the agreement between the experimental and FE analysis results. It was found that the stiffness of the specimens was on the verge of that in the experimental result even though the ultimate moment capacity of the connection obtained from the FE model for specimen was slightly higher than that from the experiments. This is because the imperfections from material and fabrication are not considered by the FE model [1]. Moreover, due to the assembly defect between rivets and columns and the small applied load at the initial stage of the test, there is a deviation between the two M-θ curves near the origin. According to formulas (2) and (3), the stiffness values of M-θ are calculated, and it is found that the stiffness values are not much different, and the average error of the four groups is about 4.6%.

During the mechanical performance test and finite element simulation, three failure modes of beam-column joints were observed. The failure modes are shown in Table 4: (i) yielding of the beam-end-connectors, (ii) tearing of the column material, especially the holes, and (iii) fracture or yielding of the rivets. When the beam-column joint is under
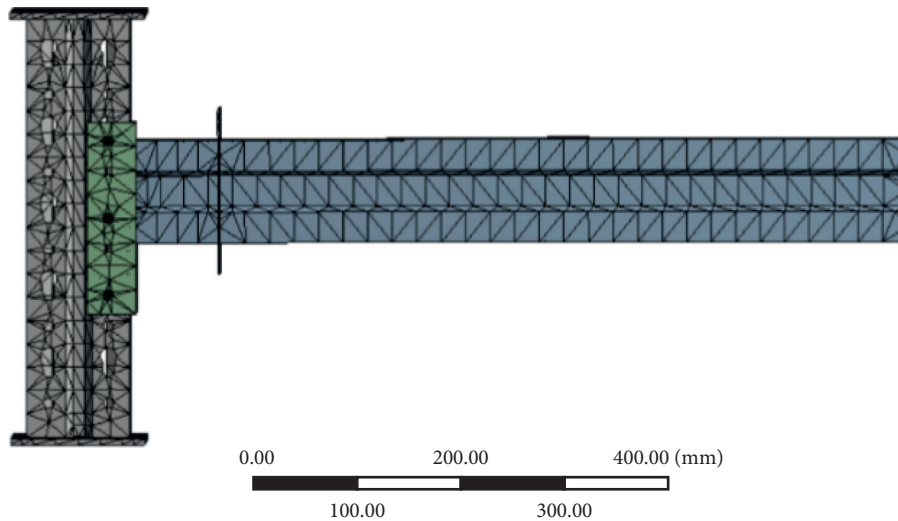
FIGURE 9: Diagram of mesh.

(a)   (b)
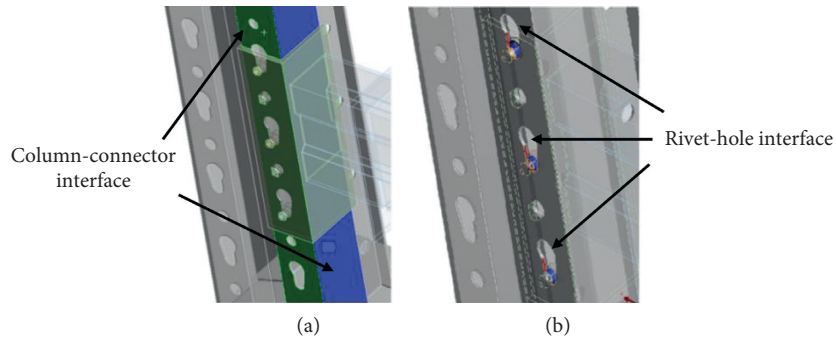
FIGURE 10: (a) The interaction between surface of column and surface of beam-end-connectors. (b) The interaction between surface of rivet and holes of column.
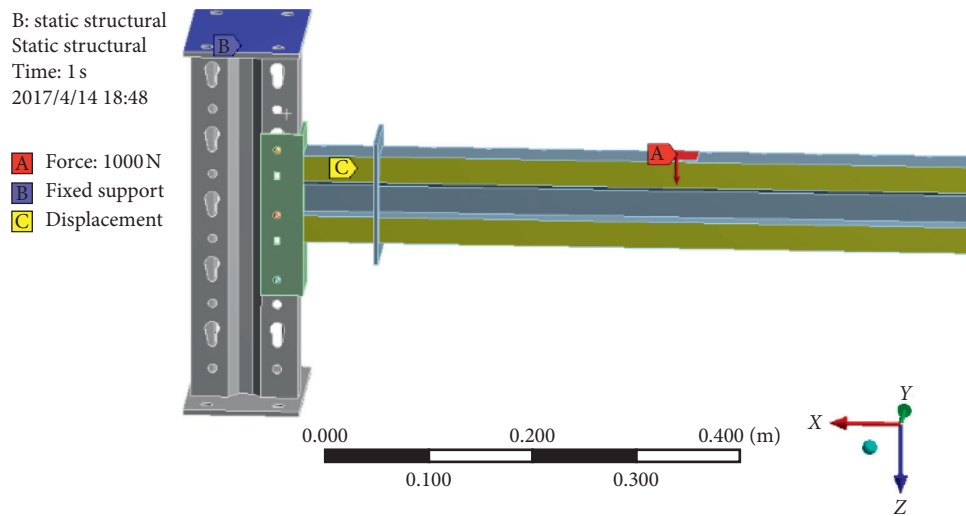
FIGURE 11: Boundary condition and loading.

compression, the rivet will bear the shear force opposite to the direction of the connector and the column, resulting in plastic deformation. The three simulated failure modes basically agree with the mechanical performance test, which verifies the validity of the simulation results. It is shown from the above diagrams that the FE model can predict the
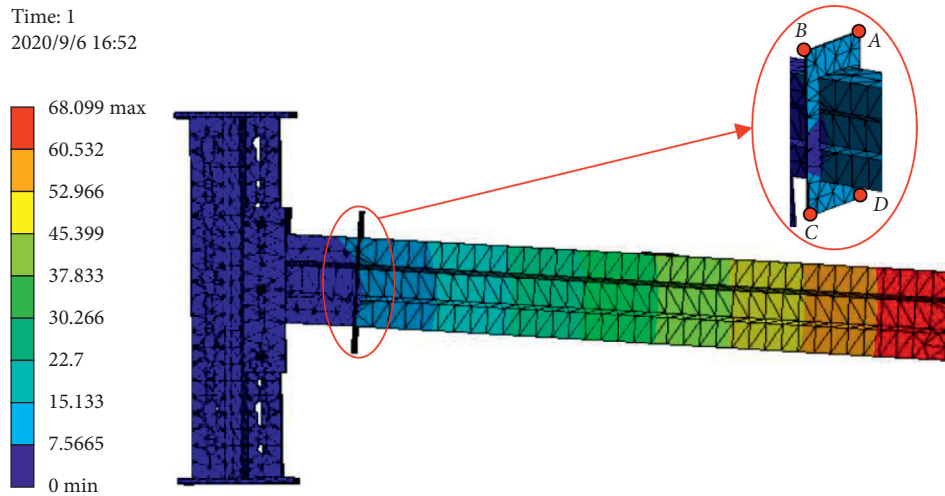
FIGURE 12: The result of finite element simulation.
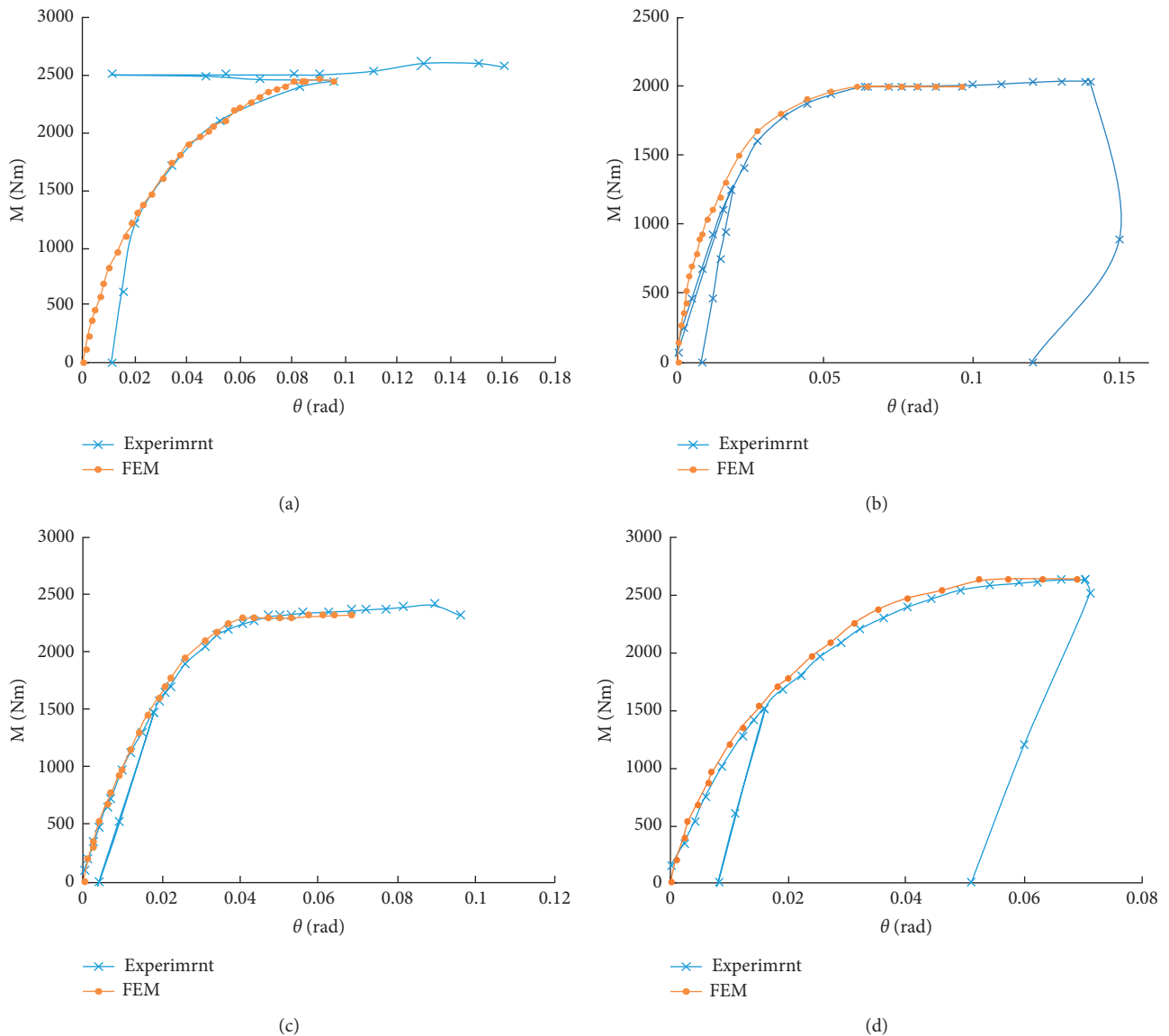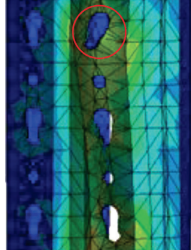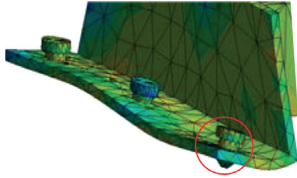
FIGURE 13: Comparison between experiment and finite element analysis. (a) Column: 90/77/2.0; beam/connector: 100/48-3 rivets. (b) Column: 100/90/2.5; beam/connector: 100/48-3 rivets. (c) Column: 100/100/2.0; beam/connector: 100/48-3 rivets. (d) Column: 120/94/2.0; beam/connector: 100/48-3 rivets.

TABLE 4: Comparison of failure modes between the experimental test and FEM.

Failure mode	Experiment	FEM
Yielding of the connectors		
Tearing of the column		
Fracture of the rivets		

experimental behavior well as a whole, and the physical test on the BCCs can be replaced by the FE simulation from the perspective of engineering structure design. Consequently, with similar FE simulation method, 432 bending tests of beam-to-column joints were carried out by selecting representative columns including M90 B, M100 A, and M100 B, which are the most widely used in industrial application. The detailed parameters and specific simulation data are shown in the Supplementary Material (available here).

4. Empirical Studies

To validate prediction performances on the SPR BCCs used by the proposed data-driven methods, some empirical cases are conducted in this study. This section first argues the mapping relationship of predictive model between the input data and output data. Then, it describes how the empirical cases were carried out. Finally, it provides a description of how the search for the parameter that achieves the best possible performance was made.

5. SVM Regression Algorithms

Support vector machines (SVMs) are based on principles of convex optimization and statistical learning theory proposed by Vapnik and Izmailov [36]. The main idea of the SVM regression algorithm is to estimate the output variable y from original input data vector x mapped into a higher-dimensional feature space through nonlinear

transformation, and extract the information and regularity contained among the data. The SVM regression function is defined as

$$y = f(x) = w^T \varphi(x) + b, \tag{3}$$

where $f(x)$ denotes the estimated value, w the weight vector, b a constant known as "bias", and $\varphi(x)$ a mapping function that maps the input data vector x into a high-dimensional feature space. Minimized risk function can avoid overfitting and thereby improving the generalization capability while obtaining the weight vector w^T, as shown in

$$\min \frac{1}{2} w^T w + \gamma \sum_{t=1}^{T} (\zeta_t + \zeta_t^*)$$

$$\text{s.t.} \quad w^T \varphi(x_t) + b \leq \phi + \zeta_t^* \ (i = 1, 2, \ldots, T), \tag{4}$$

$$y_t - (w^T \varphi(x_t) + b) \leq \phi + \zeta_t \ (i = 1, 2, \ldots, T),$$

where γ is penalty parameter, and ζ_t and ζ_t^* are the sizes of the stated excess positive and negative deviations which are termed nonnegative "slack" variables, as shown in Figure 14. It is important to note that the feature $\varphi(x)$ need not be computed; rather, what is needed is the kernel function that has to satisfy Mercer's condition. Some of the mostly used kernels include polynomial, radial basis function (RBF), and sigmoid. In this study, RBF is used which is relatively simple and suitable for high-dimensional feature sets, as shown in
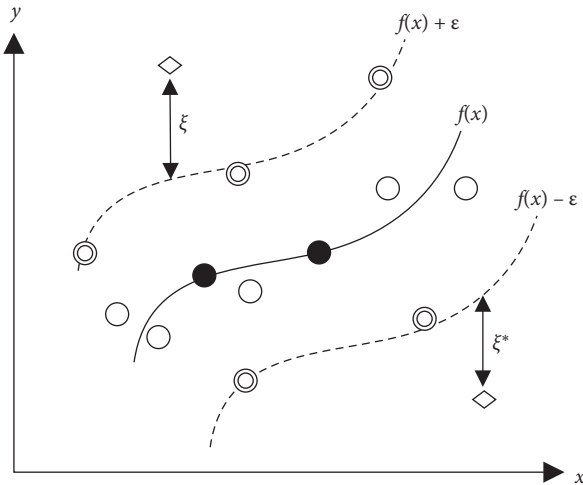
Figure 14: Diagram of support vector machine.

$$K\left(x, x_i\right) = \exp\left[\frac{-\left|x - x_i\right|^2}{2\sigma^2}\right], \tag{5}$$

where σ represents width of the RBF. Correct selection of kernel function parameters is critical to the performance and compensation accuracy of SVMs, which are described in detail in [37]. Some advances and more detailed description of SVM for regression can be found in [27, 36].

5.1. The Relational Mapping of Predictive Model Data.
The construction of the prediction model only depends on the mapping relationship between input and output data, so the determination of input and output indexes of the prediction model plays an important role in the model construction. The only output of each prediction model based on data-driven method was the bending strength (BS) of boltless BCCs in our study. According to the properties of parameters, the performance parameters that may affect the BCCs are divided into two categories: structural parameters and assembly parameters. The influence of structural parameters such as column-related parameters, beam-related parameters, and connector-related parameters is considered, respectively, as the model inputs with reference to the existing analytical model [14, 38]. On the other hand, according to the proposal in [39], the flexural rigidity of boltless BCC is largely affected by the different wielding positions between the connector and the stub beam. In fact, the boltless BCC is made up from the beam and the connector with the welding technology. The connector has an interference fit to the rivet, which makes the rivet in clearance fit with the column. Accordingly, in this paper, two additional assembly variables such as the beam position (BP) and the clearance between the connector and the column web (CL2) are also supposed to be utilized as the predictive model inputs so as to make it as close as possible to the practical application. The general relational mapping of predictive model data and its description are listed in Table 5.

5.2. Model Train

5.2.1. SVM Model Train.
In this work, the regression prediction model of support vector machine is established by compiling relevant programs using the "Regression Learner App" in Matlab2017a [33]. A total of 432 sets of data were collected from finite element simulations of beam-column joints, of which 400 sets were used for model establishment and the remaining 32 sets were used for model result verification. The establishment of support vector machine prediction model includes the selection of verification methods, kernel functions, loss accuracy, and related parameters.

(1) Choose Verification Scheme. Choose a validation method to examine the predictive accuracy of the fitted models. Validation estimates model performance on new data, helps choose the best model, and protects against overfitting. A model that is too flexible and suffers from overfitting has a worse validation accuracy. Choosing a validation scheme before training any models can allow comparing all the models in the session using the same validation scheme. Each round of cross-validation involves randomly partitioning the original dataset into a training set and a testing set. The training set is then used to train a supervised learning algorithm, and the testing set is used to evaluate its performance. This process is repeated several times, and the average cross-validation error is used as a performance indicator. This paper uses the K-fold validation method provided in the "Regression Learner App."

(2) Selection of Kernel Function. The app can train regression support vector machines (SVMs) in Regression Learner. For greater accuracy on low-through medium-dimensional datasets, train an SVM model using "fitrsvm" function. The predictive results applying different core functions including Linear SVM, Quadratic SVM, Cubic SVM, Fine Gaussian SVM, Medium Gaussian SVM, and Coarse Gaussian SVM are shown in Table 6. By comparison with the root mean square error (RMSE), R-squared, mean squared error (MSE), and mean absolute error (MAE), the Cubic SVM is selected as the best core function of the model.

(3) Advanced SVM Options. The remaining advanced SVM options in Matlab App are demonstrated as follows. The box constraint is set to 4500, the epsilon to 450, and the kernel scale to 2. The model training results are shown in Figure 16.

After training a regression model, the predicted vs. simulated response plot (as shown in Figure 17) is used to check model performance, which is used to understand how well the regression model makes predictions for different response values. A perfect regression model has a predicted response equal to the true response, so all the points lie on a diagonal line. The vertical distance from the line to any point is the error of the prediction for that point. A good model has small errors, and so the predictions are scattered near the line.

The residual plot (as shown in Figure 18) is used to check model performance. The residual plot displays the difference between the predicted and simulated responses. Usually a good model has residuals scattered roughly symmetrically around zero.
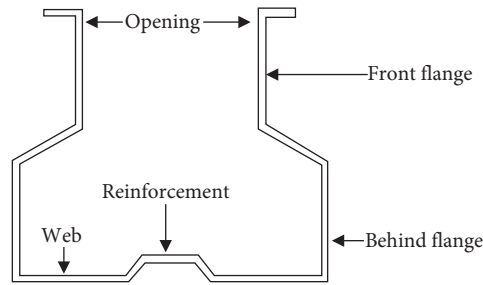
FIGURE 15: Section details of column.

TABLE 5: The input feature indicators.

Related category	Indicators	Descriptions	Notes
Column	CT	The types of the columns	On the basis of research by Shah et al. [14, 38],
	CT1	The column thickness	as shown in Figure 15
Beam	BH	The beam height	Referring to the research by Zhao et al. [11],
	BW	The beam width	the beam structure can be simplified to a rectangle for the
	BT	The beam thickness	convenience of modeling and analysis, as shown in Figure 4
Connector	OPC	The opening position of connector	On the basis of research by Markazi et al. [39],
	NT	The number of the rivets	as shown in Figure 4
	CT2	The thickness of connector	
Assembly parameters	BP	The beam position	This paper proposed assembly parameters as input indicators,
	CL	The clearance between the connector and the column web	as shown in Figure 5

TABLE 6: Comparison of prediction models of different kernel functions.

Name	RMSE	R-squared	MSE	MAE
Linear SVM	7306.48	0.21	53422823.02	4186.19
Quadratic SVM	4597.41	0.68	21327775.39	3911.93
Cubic SVM	1227.21	0.98	1522405.96	847.19
Fine Gaussian SVM	6099.04	0.45	37151937.61	4036.13
Medium Gaussian SVM	4577.70	0.69	20947018.98	2672.64
Coarse Gaussian SVM	7490.49	0.17	56083875.77	4300.80
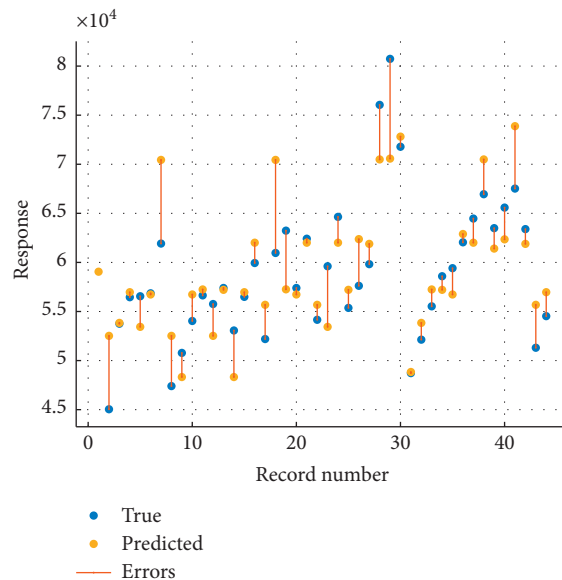
FIGURE 16: Response plot.
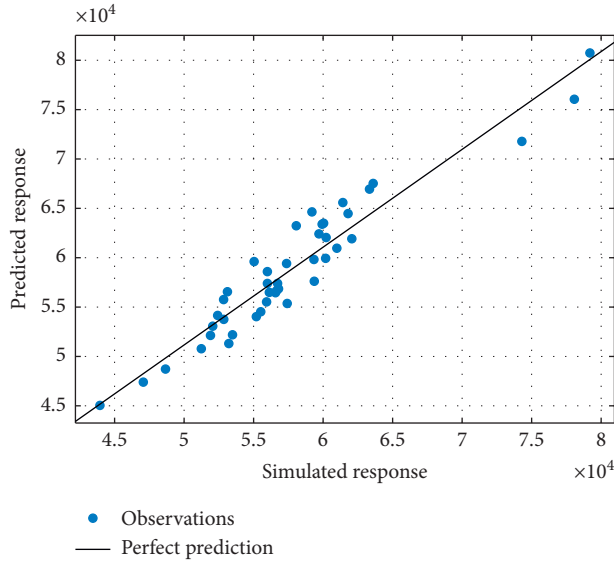
Figure 17: Predicted vs. simulated response plot.
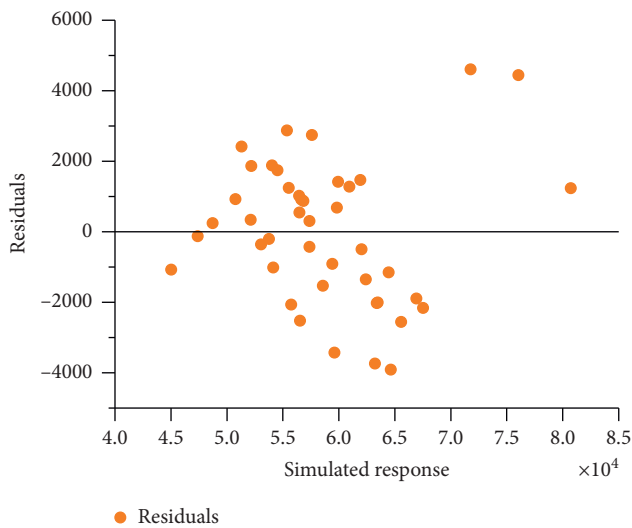
Figure 18: Residual plot.

5.2.2. ANN Model Train.

Artificial neural network (ANN) is a powerful data-modeling tool that is able to capture and represent any kind of input-output relationships. BP is widely used in engineering because of its simple model and high prediction accuracy [40].

The design of BP model program adopts the artificial neural network app in Matlab2017a. The model training uses the same 400 sets of training data as SVM model. Similarly, the datasets from BCCs virtual test are divided into two groups of training data and testing data. The training process of the model includes the determination of hidden layer number, the selection of transfer function, and the preset number of neurons in hidden layer. The error of observation results can be modified by adjusting the above parameters, until the expected results are obtained.

Table 7: The selection of ANN model parameters.

Name	Parameters
Hidden layer node number	8
Hidden layer number	1
Transfer function	logsig

Table 8: Comparison between the test values, SVM predicted values, and ANN predicted values.

Number	Test	SVM	AE (%)	ANN	AE (%)
1	34870.00	35033.05	0.47	36607.54	4.98
2	49890.00	51735.48	3.70	50863.15	1.95
3	60640.00	60463.71	0.29	62736.40	3.46
4	65170.00	65553.39	0.59	64236.76	1.43
MAE		40.7625		434.385	
RMSE		950.06		1519.20	
MAPE		0.12%		1.19%	
Correlation coefficient R		0.9978		0.9959	
R-squared		0.9956		0.9919	
Cases with an error of more than 3%		1		2	

Note. The unit of bending strength is kN mm/rad. $AE = |(\text{Test} - \text{Predicted})/x_0| \times 100\%$. $\text{MAE} = (1/N) \sum_{n=1}^{N} \text{AE}_n$

In the light of Bishop's report, more than one hidden layer is usually not necessary. Therefore, the ANN architecture for thin-walled steel design has only one hidden layer. As proposed in the literature [41], the node number of hidden layers was obtained as 8 by

$$s = \sqrt{0.43mn + 0.12n^2 + 2.54m + 0.77n + 0.35 + 0.51}, \tag{6}$$

where m is the number of neurons in the input layer and n is the number of neurons in the output layer.

The BP network toolbox in Matlab2017a has a variety of transfer functions for modeling, including linear function, nonlinear function, and other error surface functions. In this paper, "logsig" is selected as the model transfer function, which is a differentiable logarithmic s-type transfer function, which maps the input range of neurons $(-\infty, +\infty)$ to the interval of $(0, +1)$, and its equation is

$$\text{logsig}(n) = \frac{1}{1 + e^{-n}}. \tag{7}$$

The detailed parameter settings of the ANN prediction model are summarized in Table 7.

5.3. Results and Discussion.

After the establishment of the model, it still needs to be verified, so it is compared with the results of the four groups of mechanical properties tests, as shown in Table 8. It can be seen that both SVM model and BP model are close to the test value, with mean absolute error (AE) of over 3% and correlation coefficient R close to 1. The accuracy of the model is preliminarily verified. Because the training set of the prediction model comes from the calculation results of the numerical model, the overall value

TABLE 9: Comparison between the FEM values, SVM predicted values, and ANN predicted values.

Number	FEM	SVM	AE (%)	ANN	AE (%)
1	43945.22	45042.03	2.50	48907.54	11.29
2	52853.68	53755.41	1.71	52873.51	0.04
3	56572.25	56453.72	0.21	56736.45	0.29
4	53116.89	56553.71	6.47	54236.05	2.11
5	56792.16	56853.3	0.11	57766.89	1.72
6	62064.84	61920.47	0.23	62083.12	0.03
7	47064.88	47399.29	0.71	45548.75	3.22
8	51237.39	50781.86	0.89	49606.38	3.18
9	55195.33	54033.37	2.11	54038.4	2.10
. . .					
31	51897.03	51306.61	3.59	50754.98	4.63
32	55933.86	54525.76	1.78	55121.01	0.71
MAE		25.51		115.40	
RMSE		2329.36		1806.92	
MAPE		0.06%		0.24%	
Correlation coefficient R		0.9560		0.9651	
R-squared		0.9140		0.9315	
Cases with an error of more than 5%		3		7	

Note: the unit of bending strength is kN mm/rad.

is higher than the experimental value. The final results are listed in Table 9, where "FEM," "SVM," and "ANN" refer to the finite element values, the SVM predicted values, and ANN predicted values, respectively. Statistical parameters, such as the mean absolute error (%) and correlation coefficient R between the expected and real value, are used to judge the predictive power of the data-driven models. It is evident that the accuracy of all the predictive models is relatively high ($R > 95\%$), while the SVM model, in terms of the mean absolute error and the ratio of the cases with more than 5% error, is lower than the ANN model. It is evident from Table 9 that the predictive power of the SVM model the predictive power of the SVM model is the better of the two models considered here.

6. Conclusion

Due to computational complexity and accuracy, the analytical expressions for the moment-rotation stiffness of thin-walled steel beam-to-column connections are not widely used for steel member design so far. In this paper, the M-θ behavior predictions from a novel data-driven model with the integrated experimental-FEM-SVM methodology are compared with those obtained from the traditional FEM and ANN model. It is noted that the data-driven model based on SVM technique is very efficient because the prediction performance is much closer to the physical test and FEM than those obtained from the ANN models. Here, we only demonstrate that, trained with the engineering datasets from experiment and simulation, the data-driven model is able to predict the M-θ behavior of different BCCs through self-learning, which can help engineers to make quick and effective decisions for complicated rack design. The results of our paper appear to be preliminary and limited to boltless BCCs situations, but it has been found that data-driven models for solving complex semirigid component design problems are very promising. Future research should

focus on the following aspects: (1) expansion of the engineering analysis database to improve the flexibility of the data-driven model and then optimize the design configuration among a large number of beam-to-column joints; (2) development of new methodologies that can effectively explain the results of these apparently incomprehensible models. We believe that this research can be finally fused together with other pioneering analytic or experimental studies. With advancement of data mining and cloud computing techniques, many of the producers' subjective intuitions in steel pallet rack industry will finally be replaced by smart and friendly expert systems in the near future.

Data Availability

The detailed simulation data used to support the findings of this study are included within the supplementary data file. These data from FEM and tests are real and valid. They can be provided free of charge for those who need them.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

Supplementary Materials

The detailed simulation and test data used to support the results of this study are provided in the Supplementary Material (*Supplementary Materials*)

References

[1] Đ. Jovanović, D. Žarković, V. Vukobratović, and Z. Brujić, "Hysteresis model for beam-to-column connections of steel storage racks," *Thin-Walled Structures*, vol. 142, pp. 189–204, 2019.

[2] S. N. R. Shah, N. H. Ramli Sulong, R. Khan, M. Z. Jumaat, and M. Shariati, "Behavior of industrial steel rack connections," *Mechanical Systems and Signal Processing*, vol. 70-71, pp. 725–740, 2016.

[3] R. Sousa, N. Batalha, and H. Rodrigues, "Numerical simulation of beam-to-column connections in precast reinforced concrete buildings using fibre-based frame models," *Engineering Structures*, vol. 203, Article ID 109845, 2020.

[4] EN 15512, *Steel Static Storage Systems*, European Technical Committee CEN/TC 344, European Specifications, Brussels, Belgium, 2009.

[5] RMI, "Specification for the design, testing and utilization of industrial steel storage racks, ANSI MH16.1-2012," Technical Report, Rack Manufacturers Institute, Charlotte, NC, USA, 2012.

[6] AS/NZS 4084, *Steel Storage Racking*, Standards Australia/Standards New Zealand, Sydney, Australia, 2012.

[7] M. Shariati, M. M. Tahir, T. C. Wee et al., "Experimental investigations on monotonic and cyclic behavior of steel pallet rack connections," *Engineering Failure Analysis*, vol. 85, pp. 149–166, 2018.

[8] A. Kanyilmaz, C. A. Castiglioni, G. Brambilla, and G. P. Chiarelli, "Experimental assessment of the seismic behavior of unbraced steel storage pallet racks," *Thin-Walled Structures*, vol. 108, pp. 391–405, 2016.

[9] P. A. Król, M. Papadopoulos-Woźniak, and J. Wójt, "Experimental investigation on shear strength of hooking-type beam-to-column joints, applied in one of high storage pallet racking systems," *Procedia Engineering*, vol. 91, pp. 232–237, 2014.

[10] P. Prabha, V. Marimuthu, M. Saravanan, and S. Arul Jayachandran, "Evaluation of connection flexibility in cold formed steel racks," *Journal of Constructional Steel Research*, vol. 66, no. 7, pp. 863–872, 2010.

[11] X. Zhao, T. Wang, Y. Chen, and K. S. Sivakumaran, "Flexural behavior of steel storage rack beam-to-upright connections," *Journal of Constructional Steel Research*, vol. 99, pp. 161–175, 2014.

[12] K. M. Bajoria, K. K. Sangle, and R. S. Talicotti, "Modal analysis of cold-formed pallet rack structures with semi-rigid connections," *Journal of Constructional Steel Research*, vol. 66, no. 3, pp. 428–441, 2010.

[13] M. L. Chisala, "Modelling M-θ curves for standard beam-to-column connections," *Engineering Structures*, vol. 21, no. 12, pp. 1066–1075, 1999.

[14] X. Zhao, L. Dai, T. Wang, K. S. Sivakumaran, and Y. Chen, "A theoretical model for the rotational stiffness of storage rack beam-to-upright connections," *Journal of Constructional Steel Research*, vol. 133, pp. 269–281, 2017.

[15] F. Gusella, M. Orlando, and K. Thiele, "Evaluation of rack connection mechanical properties by means of the

component method," *Journal of Constructional Steel Research*, vol. 149, pp. 207–224, 2018.

[16] I. Yaqoob, I. A. T. Hashem, A. Gani et al., "Big data: from beginning to future," *International Journal of Information Management*, vol. 36, no. 6, pp. 1231–1247, 2016.

[17] H. Salehi and R. Burgueño, "Emerging artificial intelligence methods in structural engineering," *Engineering Structures*, vol. 171, pp. 170–189, 2018.

[18] W.-Q. Song, C. Cattani, and C.-H. Chi, "Fractional Brownian motion and quantum-behaved particle swarm optimization for short term power load forecasting: an integrated approach," *Energy*, vol. 194, Article ID 116847, 2019, In press.

[19] Z. M. Yaseen, M. T. Tran, S. Kim, T. Bakhshpoori, and R. C. Deo, "Shear strength prediction of steel fiber reinforced concrete beam using hybrid intelligence models: a new approach," *Engineering Structures*, vol. 177, pp. 244–255, 2018.

[20] F. Sena Cardoso and K. J. R. Rasmussen, "Finite element (FE) modelling of storage rack frames," *Journal of Constructional Steel Research*, vol. 126, pp. 1–14, 2016.

[21] Y.-J. Cha, W. Choi, and O. Büyüköztürk, "Deep learning-based crack damage detection using convolutional neural networks," *Computer-Aided Civil and Infrastructure Engineering*, vol. 32, no. 5, pp. 361–378, 2017.

[22] Z. Wang and Y. Cha, "Automated damage-sensitive feature extraction using unsupervised convolutional neural networks," in *Proceedings of the Sensors and Smart Structures Technologies for Civil, Mechanical, and Aerospace Systems 2018*, vol. 10598, Article ID 105981J, March 2018.

[23] A. Santos, E. Figueiredo, M. F. M. Silva, C. S. Sales, and J. C. W. A. Costa, "Machine learning algorithms for damage detection: kernel-based approaches," *Journal of Sound and Vibration*, vol. 363, pp. 584–599, 2016.

[24] R. Langone, E. Reynders, S. Mehrkanoon, and J. A. K. Suykens, "Automated structural health monitoring based on adaptive kernel spectral clustering," *Mechanical Systems and Signal Processing*, vol. 90, pp. 64–78, 2017.

[25] R. Ghiasi and M. R. Ghasemi, "An intelligent health monitoring method for processing data collected from the sensor network of structure," *Steel & Composite Structures*, vol. 29, pp. 703–716, 2018.

[26] V. N. Vapnik, *Statistical Learning Theory*, Wiley, New York, NY, USA, 2nd edition, 1998.

[27] P. Debnath and A. K. Dey, "Prediction of bearing capacity of geogrid-reinforced stone columns using support vector regression," *International Journal of Geomechanics*, vol. 18, no. 2, Article ID 04017147, 2018.

[28] W. Chen, J. Zhang, M. Gao, and G. Shen, "Performance improvement of 64-QAM coherent optical communication system by optimizing symbol decision boundary based on support vector machine," *Optics Communications*, vol. 410, pp. 1–7, 2018.

[29] F. De Boissieu, B. Sevin, T. Cudahy et al., "Regolith-geology mapping with support vector machine: a case study over weathered Ni-bearing peridotites, New Caledonia," *International Journal of Applied Earth Observation and Geoinformation*, vol. 64, pp. 377–385, 2018.

[30] J. Luo, X. Yan, and Y. Tian, "Unsupervised quadratic surface support vector machine with application to credit risk assessment," *European Journal of Operational Research*, vol. 280, no. 3, pp. 1008–1017, 2020.

[31] K. E. Pilario, M. Shafiee, Y. Cao, L. Lao, and S.-H. Yang, "A review of kernel methods for feature extraction in nonlinear process monitoring," *Processes*, vol. 8, no. 1, p. 24, 2019.

[32] A. Kozłowski and L. Ślęczka, "Preliminary component method model of storage rack joint," in *Proceedings of the Connections in Steel Structures V*, pp. 253–262, Amsterdam, Netherlands, June 2004.

[33] MathWorks, *MATLAB (R2017a) User Guide*, MathWorks, Natick, MA, USA, 2017, http://www.mathworks.com/support.

[34] N. Talebian, B. P. Gilbert, N. Baldassino, and H. Karampour, "Factors contributing to the transverse shear stiffness of bolted cold-formed steel storage rack upright frames with channel bracing members," *Thin-Walled Structures*, vol. 136, pp. 50–63, 2019.

[35] EN 10002-1, *Metallic Materials–Tensile Testing–Part 1: Method of Test at Ambient Temperature*, European Committee for Standardization (CEN), Brussels, Belgium, 2001.

[36] V. Vapnik and R. Izmailov, "Knowledge transfer in SVM and neural networks," *Annals of Mathematics and Artificial Intelligence*, vol. 81, no. 1-2, pp. 3–19, 2017.

[37] L. Olatomiwa, S. Mekhilef, S. Shamshirband, K. Mohammadi, D. Petković, and C. Sudheer, "A support vector machine-firefly algorithm-based model for global solar radiation prediction," *Solar Energy*, vol. 115, pp. 632–644, 2015.

[38] S. N. R. Shah, N. H. Sulong-Ramli, N. H. Sulong, R. Khan, and M. Z. Jumaat, "Structural performance of boltless beam end connectors," *Advanced Steel Construction*, vol. 13, no. 2, pp. 144–159, 2017.

[39] F. D. Markazi, R. G. Beale, and M. H. R. Godley, "Numerical modelling of semi-rigid boltless connectors," *Computers & Structures*, vol. 79, no. 26–28, pp. 2391–2402, 2001.

[40] A. K. Abbas, N. A. Al-haideri, and A. A. Bashikh, "Implementing artificial neural networks and support vector machines to predict lost circulation," *Egyptian Journal of Petroleum*, vol. 28, no. 4, pp. 339–347, 2019.

[41] K.-Y. Dong and W.-D. Yu, "A worsted yarn virtual production system based on BP neural network," *Journal of Donghua University*, vol. 21, no. 4, pp. 34–37, 2004.

Research Article

Shear Strength of Internal Reinforced Concrete Beam-Column Joints: Intelligent Modeling Approach and Sensitivity Analysis

De-Cheng Feng [1] **and Bo Fu** [2,3]

[1]*Key Laboratory of Concrete and Prestressed Concrete Structures of the Ministry of Education, Southeast University, Nanjing 211189, China*
[2]*School of Civil Engineering, Chang'an University, Xi'an 710061, China*
[3]*State Key Laboratory of Green Building in Western China, Xi'an University of Architecture and Technology, Xi'an 710055, China*

Correspondence should be addressed to Bo Fu; 90_bofu@chd.edu.cn

In this paper, an intelligent modeling approach is presented to predict the shear strength of the internal reinforced concrete (RC) beam-column joints and used to analyze the sensitivity of the influence factors on the shear strength. The proposed approach is established based on the famous boosting-family ensemble machine learning (ML) algorithms, i.e., gradient boosting regression tree (GBRT), which generates a strong predictive model by integrating several weak predictors, which are obtained by the well-known individual ML algorithms, e.g., DT, ANN, and SVM. The strong model is boosted as each weak predictor has its own weight in the final combination according to the performance. Compared with the conventional mechanical-driven shear strength models, e.g., the well-known modified compression field theory (MCFT), the proposed model can avoid the complicated derivation process of shear mechanism and calibration of the involved empirical parameters; thus, it provides a more convenient, fast, and robust alternative way for predicting the shear strength of the internal RC joints. To train and test the GBRT model, a total of 86 internal RC joint specimens are collected from the literatures, and four traditional ML models and the MCFT model are also employed as comparisons. The results indicate that the GBRT model is superior to both the traditional ML models and MCFT model, as its degree-of-fitting is the highest and the predicting dispersion is the lowest. Finally, the model is used to investigate the influences of different parameters on the shear strength of the internal RC joint, and the sensitivity and importance of the corresponding parameters are obtained.

1. Introduction

Reinforced concrete (RC) beam-column joint or connection is one of the most critical and vulnerable components in RC structures. The failure of the RC beam-column joints could seriously affect the overall safety of the structures. Especially, it will suffer from the shear failure if there are insufficient transverse reinforcements and/or the material properties are deteriorated due to the aging effects. As it is known to all, shear failure is a brittle failure type without any warnings. Therefore, it is vital to accurately predict the shear strength of the RC beam-column joints to avoid shear failure in design procedures in order to ensure the safety of the structures.

In general, there are three commonly used approaches to assess shear strength of the RC joints, i.e., experimental study, numerical simulation, and theoretical analysis. The experimental study is the most direct and classical way, which can be traced back to 1970s [1]. However, it is costly in both time and money and difficult to operate. The numerical simulation, e.g., finite element method (FEM), is also widely adopted for its low cost [2, 3]. Nevertheless, it usually has several simplifications and some of the mechanisms are hard to be reflected in the FEM framework, e.g., multistress state behavior, shear behavior, and interfacial bond-slip behavior. Apart from the experimental and numerical studies, numerous theoretical models were also proposed to evaluate the performance of the RC beam-column joints, for instance,

the well-known modified compression field theory (MCFT) [4], the strut-and-tie method (STM) [5], etc. These models are actually derived based on the shear mechanisms of fundamental RC elements and can be widely used to evaluate the behavior of any type of shear-dominated RC members, including the beam-column joints [6]. A detailed review of the theoretical and empirical models for the RC joints can be found in [7].

In recent five years, there are some latest development on RC joint models. Eom et al. [8] developed an energy-based hysteresis model for RC beam-column joints by using the energy function and the existing backbone curve of ASCE/ SEI 41-06 [9]. Hwang et al. [10] proposed a shear strength degradation model for performance-based design of interior beam-column joints. In the model, all possible failure mechanisms of beams and joints, including flexural yielding of the beam end, diagonal cracking and concrete crushing in the joint panel, bar bond-slip, and bar elongation, are considered. Later, Hwang and Park [11] developed design equations of the joint shear strength and hoop requirement for the performance-based design of interior RC beam-column joints by considering the diagonal strut mechanism and truss mechanism. The target drift ratio and bar bond parameters are defined as the requirements of the joint shear strength and hoop strength. More recently, Hwang and Park [12] modified the shear strength degradation model for interior RC joints and applied it to exterior RC joints with standard hooked bars. Hwang et al. [13] simplified the softened strut-and-tie model to facilitate design practice for the strength prediction of discontinuity regions such as the RC beam-column joints. The shear-resisting mechanisms as suggested by the softened strut-and-tie model are considered in the simplified model. Similarly, Huang and Kuang [14] proposed a shear strength model for exterior RC wide beam-column joints by introducing the softened strut-and-tie concept. Hassan and Moehle [15] collected a database of exterior and corner beam-column joints without transverse reinforcement. Based on the database, they evaluated several existing shear strength models and developed a strut-and-tie model based on the ACI 318 [16] strut-and-tie modeling provisions and an empirical model by considering the effects of joint aspect ratio, column axial load, and concrete compressive strength.

Although the above empirical or theoretical approaches offer simple and clear explanation of the shear mechanism, they also introduce empirical assumptions which will reduce their accuracy. Moreover, the derivations seem to be complicated since the iteration process is likely involved and some of the parameters are empirical that needed to be determined according to the users' experience.

In recent years, with the flourishment of artificial intelligence (AI), a brand new way is come to people's horizons, i.e., using machine learning (ML) techniques to predict the shear strength of the RC beam-column joints. ML is a type of AI, which has various functions, e.g., classification, regression, and clustering. ML can learn the characteristics of a certain type of data according to the existing database and then classify, summarize, and predict the things of interest. Prediction of the shear strength of the RC joints is

essentially a regression problem. There are already some successful applications of prediction using ML in structural engineering, for instances, evaluating the cement strength via fuzzy logic, artificial neutral network (ANN), and gene expression programming (GEP) [17, 18], modeling the concrete properties via ANN and support vector machine (SVM) [19–23], simulating the failure of brittle anisotropic materials such as masonry via ANN [24, 25], predicting the structural member capacities via hybrid ML algorithms [26, 27], detecting the structural damage via GEP [28, 29], etc. A detailed state-of-art of the application of ML in structural engineering was summarized in [30].

However, the majority of the ML algorithms used in the previous studies were individual-type learning algorithms such as ANN family [31], SVM family [32], and decision tree (DT) family [33]. The disadvantages of the individual-type learning algorithms are instable and with low accuracy. To improve their performance, a new type of learning algorithms known as ensemble learning algorithms has been recently proposed and successfully applied in various fields. The basic idea of the ensemble learning is to combine several weak learners generated by individual learning algorithms into a strong one. In brief, the ensemble learning algorithms are more stable and accurate compared to the individual learning algorithms [34]. There are mainly two categories of ensemble learning algorithms: bagging and boosting. For the bagging family, the weak learners are produced in parallel while they are produced in sequence for the boosting family. Theoretically, bagging is more efficient and can effectively reduce the variance of the prediction, and boosting is relatively less efficient in reducing the bias. In practice, boosting is superior to bagging in terms of accuracy for general cases. Therefore, one of the most typical boosting ensemble learning algorithms referred to as gradient boosting regression tree (GBRT) [35] algorithm is used in this study.

In this paper, we aim to develop a GBRT-based intelligent method for predicting the shear strength of the RC beam-column joints and make comparisons between the proposed data-driven model and some traditional ML-based models as well as the conventional mechanical-driven MCFT model. Firstly, some individual-type ML techniques, including linear regression (LR), SVM, ANN, and DT, are briefly reviewed. Then, the mathematical background and implementation of GBRT are introduced. Afterwards, the shear strength data of 86 internal RC beam-column joints are collected from the literature. Based on the database, the prediction results from the GBRT-based model are verified by a 10-fold validation test and compared with those from the individual-type ML models. In addition, one of the representative conventional mechanical-driven approaches, i.e., MCFT, is briefly summarized and also used as comparison with the GBRT model. Finally, sensitivity analysis of input variables is conducted for the GBRT model to quantify the influences of different parameters.

2. Review of the Traditional ML Techniques

2.1. Linear Regression (LR). Linear regression (LR) is one of the most widely used statistical analysis techniques in

determining the qualitative relationship between two or more variables. In general, the least square method is adopted to solve the LR problem. If only one independent variable and one dependent variable are considered and the relationship between them is approximately linear, then this type of regression analysis is called simple linear regression (SLR). On the contrary, if two or more independent variables are included and the relationship between the independent and dependent variables are approximately linear, then this regression analysis is called multiple linear regression (MLR). For the prediction problem considered in this study, more than two input parameters should be assigned as the independent variables, so it belongs to MLR.

2.2. Support Vector Machine (SVM). Based on the statistical learning theory proposed by Vapnik [36], the support vector machine (SVM) is an effective optimizing tool to improve the generalization performance and obtain the globally optimal and unique solution. In implementing the SVM regression, the primary goal is to minimize an upper bound of the generalization error based on the structural risk minimization. The essence of the SVM regression is to map the input variables into a high-dimensional feature space by a nonlinear mapping and then conduct linear regression in the space.

2.3. Artificial Neural Networks (ANN). The artificial neural network (ANN) is a complex information processing system composed of a huge number of interconnected processing elements (neurons) arranged in layers. It is the abstraction, simplification, and simulation of the structure and mechanism of biological nervous systems such as human brains. Just as the learning process in biological systems, the ANN involves adjustments to the synaptic connections between the neurons. When it is applied to solve engineering problems, a neural network can be a vector mapper which maps input vector(s) to an output one(s).

2.4. Decision Trees (DT). Decision tree (DT) is one of the basic classification and regression methods. The DT regression approach mainly refers to one of the binary tree structures, i.e., classification and regression tree (CART) algorithm, in which the characteristic values of internal nodes are only yes or no. The main task for CART is to divide the characteristic space into several units. Every unit has a certain output. As each node is judged by yes or no, the divided boundary is parallel to the coordinate axis. Any testing data can be fallen into a unit according to its characteristic and thus obtain its corresponding output.

3. Boosted ML Approach: Gradient Boosting Regression Tree (GBRT)

Though the abovementioned traditional ML methods have already been applied in several aspects of structural engineering, including predicting the behavior of structural

members, there still exist some drawbacks. For some cases, a "best" model may not be easily obtained using those algorithms. Meanwhile, models by different algorithms will have their own hypotheses, which may lead to great model uncertainty. Therefore, this paper employs the ensemble learning algorithms to generate the predictive model for the joint shear strength. Specifically, the boosting family gradient boosting regression tree (GBRT) is adopted. The ensemble learning method is superior to the individual learning method since it offers a powerful framework to obtain a strong estimator (or learner) by integrating several weak estimators (or learners) produced by the individual learning method, so the accuracy and robustness are both enhanced. The boosting idea is reflected in the weights of the weak learners: the one with higher score will get higher weight in the final strong learner. The fundamental and theoretical backgrounds, as well as the implementation procedure, are all presented herein.

3.1. Gradient Boosting Framework of Ensemble ML. As mentioned before, ensemble learning is not an individual-type ML method. It is accomplished by integrating multiple weak learners into a strong one. Boosting is a major group of ensemble learning algorithms, which generates the weak learners subsequently and can be interpreted as an optimization algorithm on a suitable cost function. The basic idea of boosting is to update the weight of each weak learner by its learning error. If a weak learner has a large learning error, it will be assigned a large weight so that it could be paid more attention in the subsequent training process. Like other boosting methods, the gradient boosting integrates several weak learners into a single strong learner in an iterative way.

Supposing it requires M steps to find out the final strong learner and at the m-th $(m \in [1, M])$ step we have an imperfect model $f_m(\mathbf{x})$ which is the sum of weak learners in the previous steps,

$$f_m(\mathbf{x}) = \sum_{i=1}^{m-1} \alpha_i h_i(\mathbf{x}), \tag{1}$$

where $\mathbf{x}$ is the vector containing the input variables; $h_i(\mathbf{x})$ and α_i are the weak learner and the corresponding weight at step $i \in [1, m]$.

The imperfect model can be improved by adding a new weak learner $h_m(\mathbf{x})$ as $f_{m+1}(\mathbf{x}) = f_m(\mathbf{x}) + h_m(\mathbf{x})$. Then, the optimization problem becomes how to find $h_m(\mathbf{x})$. The solution of gradient boosting starts with the observation that a perfect $h_m(\mathbf{x})$ would imply

$$f_{m+1}(\mathbf{x}) = f_m(\mathbf{x}) + h_m(\mathbf{x}) = y, \tag{2}$$

where y is the target output or the tested value of the output. Equation (2) can be equivalently expressed as

$$h_m(\mathbf{x}) = y - f_m(\mathbf{x}). \tag{3}$$

Therefore, in the following gradient boosting algorithm fits $h_m(\mathbf{x})$ with the residual $y - f_m(\mathbf{x})$. Like other members of the boosting algorithms, $f_{m+1}(\mathbf{x})$ is attempted to correct

the errors of its predecessor $f_m(\mathbf{x})$. It is observed that the residual $y - f_m(\mathbf{x})$ is the negative gradient of the squared loss function $1/2[y - f_m(\mathbf{x})]^2$, so the negative gradient can be extended to other kinds of loss functions. In other words, the gradient boosting algorithm is a gradient descent algorithm, which can be generalized by varying the loss function and the gradient.

3.2. Gradient Boosting Regression Tree (GBRT).

As can be seen in the previous section, gradient boosting is actually a framework for ensembling numerous weak learners, rather than a specific learning algorithm. Theoretically, any individual algorithms from the ANN, SVM, and DT families can be used to train the weak learners. However, unlike other boosting algorithms, the individual algorithm for training the weak learners in gradient boosting is restricted to the DT algorithms, thus it is called as GBRT. In each step (or iteration), a new DT is established by fitting the negative gradient of the loss function. The number of DT is determined by the iteration number.

The GBRT model superimposes multiple DTs and is expressed as

$$f_M(\mathbf{x}) = \sum_{m=1}^{M} T(\mathbf{x}; \Theta_m), \tag{4}$$

where $T(\mathbf{x}; \Theta_m)$ represents the weak learner by DT; Θ_m denotes the parameters of m-th DT model; M is the number of DTs, respectively.

For a dataset $D = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_N, y_N)\}$ where N denotes the number of the samples, the essence of training the boosting DT model is selecting the optimal parameters of DTs $\Theta = \{\Theta_1, \Theta_2, \ldots, \Theta_M\}$ to minimize the loss function $\sum_{i=1}^{N} L[y_i, f_M(\mathbf{x}_i)]$, i.e.,

$$\arg\min_{\Theta} \sum_{i=1}^{N} L[y_i, f_M(\mathbf{x}_i)] = \arg\min_{\Theta} \sum_{i=1}^{N} L\left[y_i, \sum_{m=1}^{M} T(\mathbf{x}; \Theta_m)\right]. \tag{5}$$

Here, the loss function is used to reflect the difference between the sample real value y_i and the output of the GBRT $f_M(\mathbf{x}_i)$.

Note that the GBRT model in equation (4) can also be written in a forward step way and expressed as

$$f_m(\mathbf{x}) = f_{m-1}(\mathbf{x}) + T(\mathbf{x}; \Theta_m), \quad m = 1, 2, \ldots, M. \tag{6}$$

Therefore, training of the GBRT model can be achieved by M iteration steps. Specifically, at the initial step, we define $f_1(\mathbf{x}) = 0$, and for the m-th iteration step, a new $T(\mathbf{x}; \Theta_m)$ is generated. The parameters Θ_m of $T(\mathbf{x}; \Theta_m)$ should be obtained to minimize the loss function

$$\widehat{\Theta}_m = \arg\min_{\Theta_m} \sum_{i=1}^{N} L[y_i, f_{m-1}(\mathbf{x}_i) + T(\mathbf{x}_i; \Theta_m)], \tag{7}$$

where $\widehat{\Theta}_m$ are the optimal DT parameters.

If the squared loss function is used, then one obtains

$$L[y_i, f_{m-1}(\mathbf{x}_i) + T(\mathbf{x}_i; \Theta_m)] = [y_i - f_{m-1}(\mathbf{x}_i) - T(\mathbf{x}_i; \Theta_m)]^2$$
$$= [r_{m,i} - T(\mathbf{x}_i; \Theta_m)]^2, \tag{8}$$

where $r_{m,i} = y_i - f_{m-1}(\mathbf{x}_i)$ represents the residual of the model $f_{m-1}(\mathbf{x})$.

Therefore, the solution of equation (8) converts to the selection of appropriate Θ_m to minimize the difference of the residual $r_{m,i}$ of the DT and the output $T(\mathbf{x}; \Theta_m)$ or, equivalently, $\{(\mathbf{x}_i, r_{m,i})\}_{i=1,2,\ldots,N}$ can be used as the sample set of the decision tree $T(\mathbf{x}; \Theta_m)$, and the optimal parameters $\widehat{\Theta}_m$ are obtained according to the conventional DT generation process.

Moreover, in a more generalized sense, the negative gradient of the loss function can be used to represent the residual of the model, i.e.,

$$r_{m,i} = -\left[\frac{\partial L(y_i, f(\mathbf{x}_i))}{\partial f(\mathbf{x}_i)}\right]_{f(\mathbf{x}) = f_{m-1}(\mathbf{x})}. \tag{9}$$

With $\{(\mathbf{x}_i, r_{m,i})\}_{i=1,2,\ldots,N}$, we can fit the m-th DT h_m, whose leaf nodes can be represented by $R_{m,j}, j = 1, 2, \ldots, J$, where J indicates the number of leaf nodes of the DT. For each leaf node of the regression tree h_m, calculate the optimal fitting value $c_{m,j}$:

$$c_{m,j} = \arg\min_{c} \sum_{\mathbf{x}_i \in R_{m,j}} L[y_i, f_{m-1}(\mathbf{x}_i) + c], \quad j = 1, 2, \ldots, J. \tag{10}$$

Then, the weak learner for this step can be written as

$$T_m(\mathbf{x}) = \sum_{j=1}^{J} c_{m,j} I(\mathbf{x} \in R_{m,j}), \tag{11}$$

and the updated strong learner till this step is

$$f_m(\mathbf{x}) = f_{m-1}(\mathbf{x}) + T_m(\mathbf{x}) = f_{m-1}(\mathbf{x}) + \sum_{j=1}^{J} c_{m,j} I(\mathbf{x} \in R_{m,j}). \tag{12}$$

After M steps, the strong learner is finally obtained by

$$f_M(\mathbf{x}) = \sum_{m=1}^{M} \sum_{j=1}^{J} c_{m,j} I(\mathbf{x} \in R_{m,j}). \tag{13}$$

The procedure of the GBRT algorithm can be summarized as follows:

(1) Initialization of the function for the weak learner $f_0(\mathbf{x}) = \arg\min_{c} \sum_{i=1}^{N} L(y_i, c)$

(2) For the m-th iteration ($m = 1, 2, \ldots, M$):

    (a) For each sample $(\mathbf{x}_i, y_i), i = 1, 2 \ldots, N$, the negative gradient is calculated using equation (9)

    (b) Train the m-th DT h_m by using $\{(\mathbf{x}_i, r_{m,i})\}_{i=1,2,\ldots,N}$, and the corresponding areas

of the $i-$th leaf nodes are denoted as $R_{m,j}, j = 1, 2, \ldots, J$

(c) For each leaf node of the regression tree h_m, calculate the optimal fitting value $c_{m,j}$ using equation (10)

(d) Update the learner $f_m(\mathbf{x}) = f_{m-1}(\mathbf{x}) + \sum_{j=1}^{J} c_{m,j} I(\mathbf{x} \in R_{m,j})$

(3) After M iterations, the strong learner is obtained using equation (13)

3.3. Implementation of GBRT. In this study, one of the most widely used DT, i.e., CART, is employed as the individual learning algorithm. The implementation of the GBRT can be summarized as the following four steps:

(1) Collect and process the data, such as the setting of input/output variables and the grouping of the training/testing datasets

(2) Train the regression model using the GBRT with the training dataset

(3) Validate the trained model with the testing dataset

(4) Apply the model to the realistic problems

The flowchart of the abovementioned procedure is depicted in Figure 1.

Another important issue associated with the implementation of GBRT is the determination of model parameters, which have two levels, i.e., the framework level and the level for the individual learning algorithm. At the framework level, there are two main parameters, i.e., the number of iteration (number of weak learners) and the learning rate, which is used to avoid the overfitting problem. At the single learning algorithm level, there are four primary parameters, i.e., the maximum depth of the tree, the minimum samples for split, the minimum samples of leaf node, and the minimum change in impurity. The selected values of these parameters are determined based on previous studies in literature and practical modeling experience, as shown in Table 1.

4. Collection of Experimental Data for Shear Strength of Internal RC Beam-Column Joints

In implementing the ML techniques for prediction of the shear strength for RC joints, an experimental database is required to train the predictive model and validate the model. Therefore, a database including the experimental results of 86 internal RC beam-column joints was collected for this purpose in this study. In the database, there are 10 input parameters covering material properties and geometric dimensions and reinforcing details of the test specimens, i.e., the concrete strength f_c, the section width of column b_c, the section height of column h_c, the section width of beam b_b, the section height of beam h_b, the yielding strength of beam longitudinal bar $f_{y,b}$, the yielding strength of column longitudinal bar $f_{y,c}$, the yielding strength of joint transverse bar f_{yv}, the transverse bar ratio ρ_v, and the axial load ratio n. The only output is the joint shear strength τ. The statistical information of these parameters, e.g., mean and standard deviation (St.D.), and the distributions of the aforementioned parameters are illustrated in Table 2 and Figure 2. The details of the tested specimens in the database are given in Table 3.

5. Results and Discussion

5.1. 10-Fold Cross-Validation Results. To validate the proposed method, the 10-fold cross-validation method is firstly used to evaluate the model's performance. The 10-fold cross-validation method is developed to minimize the bias associated with random sampling of the training and testing datasets. It divides the experimental data samples into 10 subsets, and for each run, 9 are set as training subsets and 1 is set as validating subset. It is believed that repeating this for 10 times is able to represent the generalization and reliability of the predictive model. Moreover, three commonly used indicators are introduced to assess the prediction performance, which are respectively defined as

Coefficient of determination R-squared (R^2):

$$R^2 = 1 - \frac{\sum_{i=1}^{N} (P_i - T_i)^2}{\sum_{i=1}^{N} (T_i - \overline{T})^2}. \qquad (14)$$

Root mean squared error (RMSE):

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{N} (P_i - T_i)^2}{N}}. \qquad (15)$$

Mean absolute error (MAE):

$$\text{MAE} = \frac{\sum_{i=1}^{N} |P_i - T_i|}{N}, \qquad (16)$$

where P_i and T_i are the predicted and tested values, respectively; $\overline{T}$ is the mean value of all the tested values; N is the total number of the samples in the dataset.

Among the three indicators, R^2 indicates the degree of the linear correlation between the predicted and tested values. RMSE shows the deviation between the predicted and tested values. MAE reflects the ratio of the prediction error to the tested values. The closer the R^2 to 1, the smaller the RMSE or MAE, the better performance the prediction model possesses. Table 4 shows the 10-fold cross-validation statistic results of the GBRT model.

It can be drawn from Table 4 that the average determination coefficient R^2 for the 10-fold results is 0.875, which is close to 1; the average RMSE and MAE are 0.948 MPa and 0.722 MPa, respectively, which are small. The standard derivations (St.D.) for R^2, RMSE, and MAE
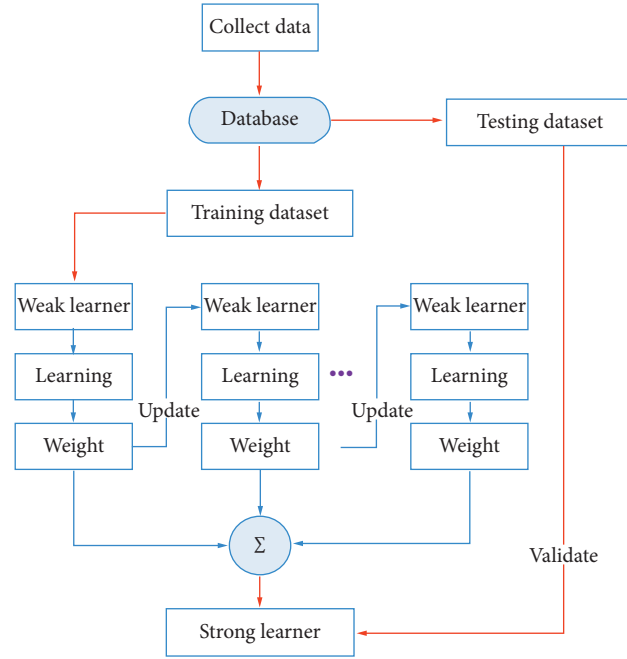
FIGURE 1: Flowchart of implementation of GBRT.

TABLE 1: Model parameters of GBRT.

Level	Parameter	Setting
Framework	Maximum iteration number	100
	Learning rate	0.03
Single learning algorithm	Maximum depth of the tree	10
	Maximum leaf nodes	200
	Minimum samples for split	6
	Minimum samples of leaf node	5

TABLE 2: Statistics of parameters for the internal RC beam-column joints.

Parameter	Unit	Maximum	Minimum	Mean	St.D.	Type
f_c	MPa	116.90	16.57	43.31	19.07	Input
b_c	mm	458.00	200.00	268.12	66.03	Input
h_c	mm	458.00	200.00	277.23	81.72	Input
b_b	mm	406.00	150.00	194.74	64.60	Input
h_b	mm	500.00	250.00	322.59	56.96	Input
$f_{y,b}$	MPa	1456.00	314.00	457.48	179.87	Input
$f_{y,c}$	MPa	1456.00	314.00	458.09	150.46	Input
f_{yv}	MPa	1374.00	276.00	451.73	262.08	Input
ρ_v	%	2.00	0.18	0.66	0.38	Input
n	—	1.18	0.00	0.25	0.23	Input
τ	MPa	14.87	2.85	7.30	2.81	Output

are 0.082, 0.347, and 0.245, respectively, which means the prediction performance has low variance. All of these indices demonstrate that the proposed method has excellent performance in predicting the shear strength of internal RC joints.

5.2. Prediction Results of Different ML Models. To demonstrate the prediction performance of the GBRT model, the four general ML models, i.e., LR, SVM, ANN, and DT, are also used to predict the shear strength of the 86 specimens. The optimized parameters of the four models are determined
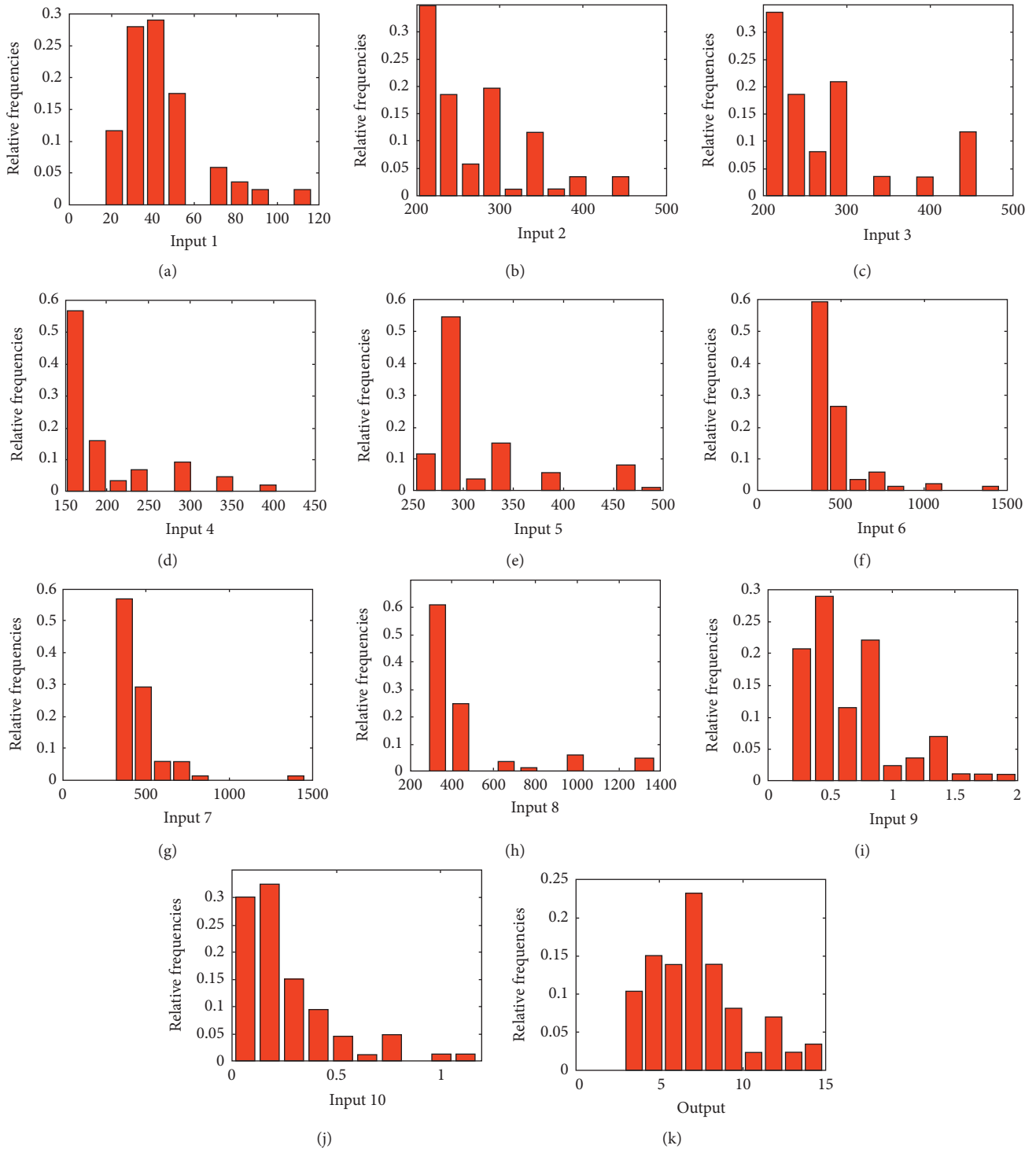
FIGURE 2: Statistic distribution of the input and output parameters. (a) Concrete strength (MPa). (b) Section width of column (mm). (c) Section height of column (mm). (d) Section width of beam (mm). (e) Section height of beam (mm). (f) Yielding strength of beam longitudinal bar (MPa). (g) Yielding strength of column longitudinal bar (MPa). (h) Yielding strength of joint transverse bar (MPa). (i) Transverse bar ratio (%). (j) Axial load ratio. (k) Joint shear strength (MPa).

Table 3: Main data of the internal RC beam-column joint tests.

References	Specimen	f_c (MPa)	$b_c \times h_c$ (mm × mm)	$b_b \times h_b$ (mm × mm)	$f_{y,b}$ (MPa)	$f_{y,c}$ (MPa)	f_{yv} (MPa)	ρ_v	n	τ (MPa)
[37]	I	24.70	250 × 250	150 × 350	391.1	353.8	277.5	0.36	0.226	4.15
	I-1	27.20	250 × 250	150 × 350	379.4	366.0	301.1	0.36	0.204	4.16
	I-2	31.95	250 × 250	150 × 350	379.4	366.0	301.1	0.36	0.173	5.10
	SJ1-1	31.78	250 × 250	150 × 350	366.3	349.0	300.9	0.42	0.173	5.53
	SJ1-2	34.48	250 × 250	150 × 350	366.3	349.0	300.9	0.42	1.180	5.82
	SJ1-4	29.80	250 × 250	150 × 350	380.3	380.3	300.9	0.42	0.440	7.32
	SJ2-2	28.40	250 × 250	150 × 350	366.3	349.0	300.9	1.50	0.174	6.17
	SJ5-1	37.88	250 × 250	150 × 350	366.3	349.0	300.9	0.42	0.178	6.47
	SJ-3	34.48	250 × 250	150 × 350	462.0	404.6	340.0	1.26	0.125	6.41
	SJ-B	27.00	450 × 450	250 × 500	326.0	388.0	384.0	0.76	0.167	5.49
[38]	J3-50	46.12	200 × 200	150 × 300	390.0	385.0	307.0	0.83	0.285	7.14
	J4-50	47.40	200 × 200	150 × 300	390.0	385.0	307.0	0.83	0.361	7.83
	J4-30	46.74	200 × 200	150 × 300	390.0	385.0	307.0	1.33	0.374	7.92
	J5-80	46.71	200 × 200	150 × 300	328.0	405.0	307.0	0.66	0.467	8.80
	J5-50	48.16	200 × 200	150 × 300	328.0	405.0	307.0	0.83	0.449	8.98
	J6-80	46.94	200 × 200	150 × 300	328.0	405.0	307.0	0.66	0.558	9.26
	J6-50	43.60	200 × 200	150 × 300	328.0	405.0	307.0	0.83	0.514	8.29
	J6-30	48.12	200 × 200	150 × 300	328.0	405.0	307.0	1.33	0.540	9.07
	J7-80	43.60	200 × 200	150 × 300	328.0	405.0	307.0	0.66	0.716	7.91
	J7-50	48.42	200 × 200	150 × 300	328.0	405.0	307.0	0.83	0.636	8.98
	J7-30	43.81	200 × 200	150 × 300	328.0	405.0	307.0	1.33	0.712	8.84
	J8-50	49.45	200 × 200	150 × 300	328.0	405.0	307.0	0.83	0.744	9.22
	J8-30	43.81	200 × 200	150 × 300	328.0	405.0	307.0	1.33	0.814	8.38
[39]	ZHJ4	27.36	200 × 200	150 × 250	360.2	360.2	350.0	1.00	0.330	6.91
	ZHJ5	27.36	200 × 200	150 × 250	360.2	360.2	350.0	1.21	0.330	6.93
	ZHJ6	27.36	200 × 200	150 × 250	360.2	360.2	350.0	1.81	0.330	7.07
[40]	J-1	27.36	200 × 200	150 × 250	380.0	380.0	350.0	1.04	0.330	6.78
	J-2	27.36	200 × 200	150 × 250	380.0	380.0	350.0	1.12	0.330	6.84
	J-3	16.57	200 × 200	150 × 250	473.0	473.0	375.0	1.30	0.270	6.49
	J-4	19.53	200 × 200	150 × 250	473.0	473.0	375.0	2.00	0.23	6.55
	JL1	21.43	250 × 250	150 × 300	314.0	314.0	276.0	0.58	0.21	5.32
[41]	SJ-3	29.34	350 × 350	175 × 350	354.7	348.0	381.5	0.50	1.06	6.88
[42]	J4	25.7	300 × 300	200 × 300	401	401	368	0.28	0.30	4.22
	J5	28.7	300 × 300	200 × 300	401	401	368	0.28	0.07	5.04
	C2	25.6	300 × 300	200 × 300	324	422	368	0.90	0.08	5.36
[43]	U1	26.2	331 × 458	280 × 458	449	457	409	0.50	0.40	5.38
	U2	41.8	331 × 458	280 × 458	449	449	409	0.50	0.25	7.87
	U3	26.6	331 × 458	280 × 458	449	402	409	0.50	0.39	6.03
	U4	36.1	458 × 331	406 × 458	449	449	409	0.50	0.29	7.17
	U5	35.9	331 × 458	280 × 458	449	449	409	0.50	0.04	7.57
	U6	36.8	331 × 458	280 × 458	449	449	409	0.50	0.48	8.15
	U7	37.2	458 × 331	406 × 458	449	449	409	0.50	0.46	7.24
[44]	J1	70.0	300 × 300	200 × 300	718	718	955	0.75	0.12	10.84
	J3	107.0	300 × 300	200 × 300	718	718	955	0.75	0.12	13.75
	J4	70.0	300 × 300	200 × 300	718	718	955	0.75	0.12	11.44
	J5	70.0	300 × 300	200 × 300	718	718	955	0.75	0.12	11.26
	J6	53.5	300 × 300	200 × 300	718	718	955	0.75	0.12	9.98
[45]	J1	81.2	300 × 300	240 × 300	638	638	1374	0.38	0.11	11.58
	J2	81.2	300 × 300	240 × 300	1456	1456	1374	0.38	0.11	12.43
	J4	72.8	300 × 300	240 × 300	515	515	1374	0.38	0.13	11.94
	J5	72.8	300 × 300	240 × 300	839	839	1374	0.38	0.13	13.38
	J8	79.2	300 × 300	240 × 300	370	370	775	0.38	0.12	14.01

TABLE 3: Continued.

References	Specimen	f_c (MPa)	$b_c \times h_c$ (mm × mm)	$b_b \times h_b$ (mm × mm)	$f_{y,b}$ (MPa)	$f_{y,c}$ (MPa)	f_{yv} (MPa)	ρ_v	n	τ (MPa)
	Aa-4	30.4	200 × 200	150 × 300	419	419	350	0.22	0.06	3.19
	Aa-7	38.1	200 × 200	150 × 300	400	400	350	0.22	0.05	3.08
	Aa-8	38.1	200 × 200	150 × 300	400	400	350	0.22	0.10	3.19
[46]	Aa-1	41.1	200 × 200	150 × 300	400	400	350	0.22	0.05	3.10
	Aa-2	41.1	200 × 200	150 × 300	400	400	350	0.22	0.10	2.97
	Ab-1	41.1	200 × 200	150 × 300	400	400	350	0.22	0.05	3.06
	Ab-2	41.1	200 × 200	150 × 300	400	400	350	0.22	0.10	2.85
	HNO3	88.7	400 × 400	300 × 400	442	442	681	0.86	0.17	13.01
[47]	HNO4	88.7	400 × 400	300 × 400	605	610	681	0.86	0.17	14.87
	HNO5	116.9	400 × 400	300 × 400	623	610	681	0.86	0.13	12.39
[48]	S3	24	300 × 300	200 × 300	465	450	390	0.36	0.05	3.64
	A1	40.2	220 × 220	160 × 250	1069	644	291	0.43	0.08	5.25
[49]	A2	40.2	220 × 220	160 × 250	409	388	291	0.43	0.08	4.87
	A3	40.2	220 × 220	160 × 250	1069	644	291	0.43	0.23	5.25
	LIJ1	25.5	343 × 457	343 × 343	440	470	400	0.61	0	3.79
[50]	LIJ2	32.8	343 × 457	343 × 343	440	470	400	0.61	0	4.86
	LIJ3	31.1	343 × 457	343 × 343	440	470	400	0.61	0	4.61
	LIJ4	34.3	343 × 457	343 × 343	440	470	400	0.61	0	5.03
	BCJ2	30.3	254 × 254	203 × 305	414	448	414	0.52	0	5.55
[51]	BCJ3	27.4	305 × 254	203 × 305	414	448	414	0.43	0	5.08
	BCJ4	27.2	356 × 254	203 × 305	414	448	414	0.37	0	5.11
	HSLCJ-1	48.6	260 × 260	150 × 300	435	421	449	0.18	0.15	7.25
[52]	HSLCJ-2	48.9	260 × 260	150 × 300	435	421	449	0.22	0.15	7.5
	HSLCJ-3	49.5	260 × 260	150 × 300	435	421	449	0.18	0.25	7.76
	HSLCJ-4	48.2	260 × 260	150 × 300	435	421	449	0.22	0.25	7.82
	JD1	41.9	250 × 250	150 × 300	427	443	318	0.33	0.147	7.28
	JD2	43.95	250 × 250	150 × 300	427	443	318	0.84	0.147	7.38
[53]	JD3	43.34	250 × 250	150 × 300	427	443	318	1.44	0.147	7.46
	JD5	41.06	250 × 250	150 × 300	427	443	318	0.84	0.059	6.35
	JD6	44.35	250 × 250	150 × 300	427	443	318	0.84	0.235	8.05
[54]	MZJD-1	37.2	300 × 300	200 × 400	320	331	318	0.67	0.1	5.85
	MZJD-2	37.2	300 × 300	200 × 400	320	331	318	0.67	0.3	7.29
[55]	8-3	54.17	250 × 250	200 × 300	479.6	479.6	479.6	0.49	0.1	9.86
	8-5	55.34	250 × 250	200 × 300	479.6	479.6	479.6	0.49	0.1	11.29

TABLE 4: 10-fold cross-validation statistic results.

Fold	R^2	RMSE (MPa)	MAE (MPa)
1	0.827	1.217	0.937
2	0.926	0.816	0.704
3	0.937	0.726	0.619
4	0.682	1.621	1.008
5	0.819	0.967	0.800
6	0.910	0.873	0.579
7	0.952	0.531	0.394
8	0.962	0.637	0.460
9	0.905	0.649	0.525
10	0.833	1.443	1.192
Average	0.875	0.948	0.722
St.D.	0.082	0.347	0.245

by using the grid search after setting the initial values. The total dataset is divided into for training and testing as 8-2, i.e., 80% of the data is used for training and 20% of the data is used for testing. Figure 3 shows the prediction results of the GBRT model and the four general ML models for the testing dataset. It is clear that the GBRT model has stronger linear correlation compared with other four ML models. The reason is that the GBRT is an ensemble learning algorithm with strong learner, while other four models use individual-type learning algorithms with weak learners.
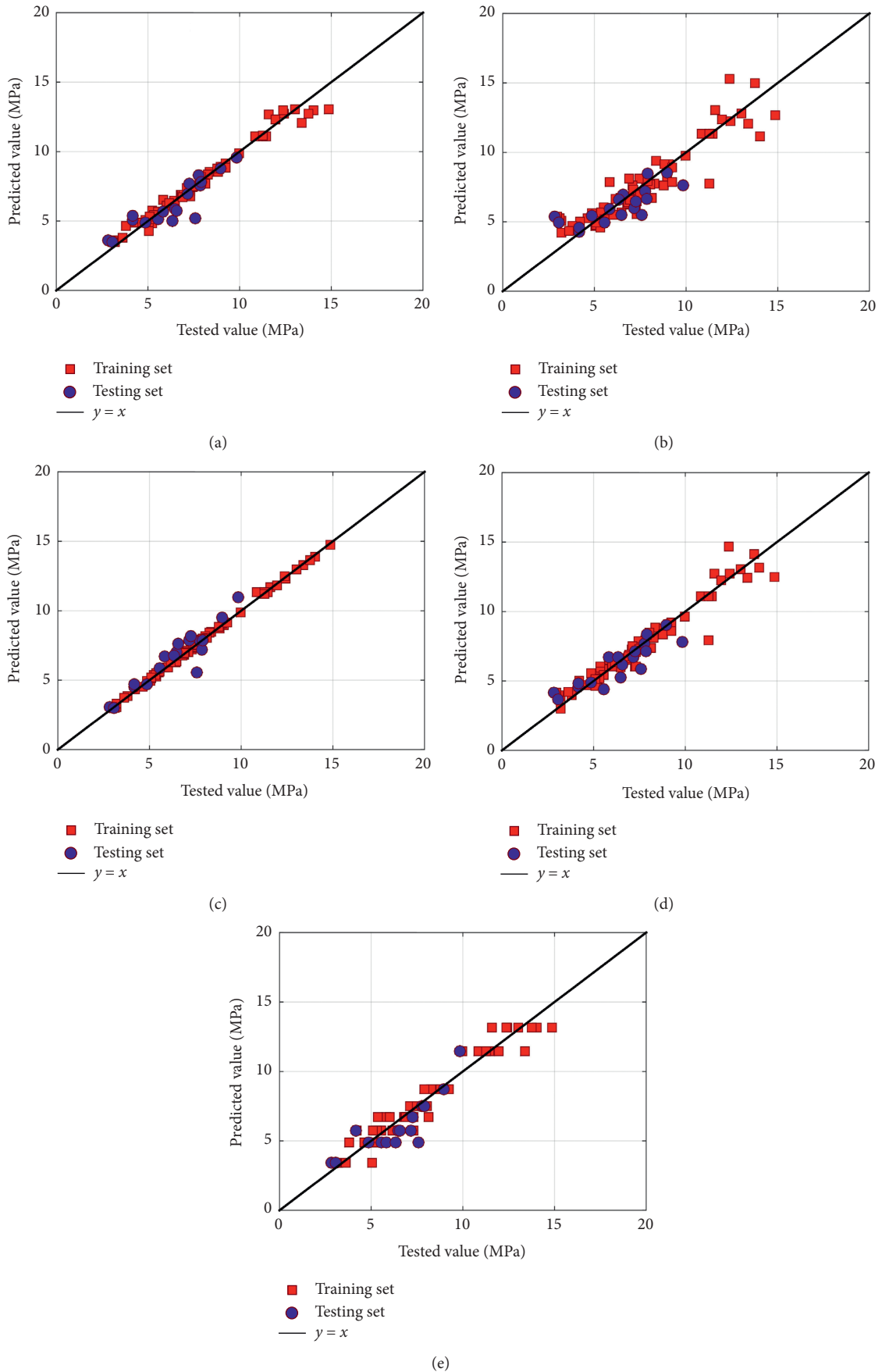
Figure 3: Prediction results of different ML models: (a) GBRT. (b) LR. (c) SVM. (d) ANN. (e) DT.

Table 5 exhibits the prediction performance of the five ML models by providing the average statistical indices of the 10-fold cross-validation results. Obviously, the GBRT model has the closest R^2 to 1 and smallest values of RMSE and MAE among the five ML models. It further verifies the superiority of the GBRT model over the general individual-type ML models.

6. Comparison with Conventional Mechanical-Driven Approach

6.1. Typical Mechanical-Driven Approach: MCFT. In this section, the derivation of MCFT is briefly summarized as it is a representative conventional mechanical-driven shear strength prediction method. A basic assumption for MCFT is that the crack direction of a RC plane element is in accordance with the principal compressive stress and varies accordingly. The definitions of stress, strain, rotational angle, and principal direction are illustrated in Figure 4, where the *x-y* coordinate system is the local system and the 1-2 coordinate system indicates the principal tensile strain-principal compressive strain system. The strain vector and stress vector of the RC element in the local system are denoted as $\begin{bmatrix} \varepsilon_x & \varepsilon_y & \gamma_{xy} \end{bmatrix}^T$ and $\begin{bmatrix} f_x & f_y & v_{xy} \end{bmatrix}^T$, respectively.

The derivation of the MCFT includes three parts, i.e., compatibility equations, equilibrium equations, and constitutive equations. The detailed formulations are given as follows.

6.1.1. Compatibility Equations. According to Mohr's circle of strain, the principal tensile strain ε_1 and the principal compressive strain ε_2 of the element are calculated as

$$\begin{cases} \varepsilon_1 = \dfrac{(\varepsilon_x + \varepsilon_y)}{2} + \sqrt{\dfrac{(\varepsilon_x - \varepsilon_y)^2}{4} + \dfrac{(\gamma_{xy})^2}{4}}, \\[4mm] \varepsilon_2 = \dfrac{(\varepsilon_x + \varepsilon_y)}{2} - \sqrt{\dfrac{(\varepsilon_x - \varepsilon_y)^2}{4} + \dfrac{(\gamma_{xy})^2}{4}}. \end{cases} \quad (17)$$

Accordingly, the rotational angle θ from the principal strain direction to the *x*-axis can be obtained by

$$\tan^2\theta = \frac{\varepsilon_x - \varepsilon_2}{\varepsilon_y - \varepsilon_2} = \frac{\varepsilon_1 - \varepsilon_y}{\varepsilon_1 - \varepsilon_x} = \frac{\varepsilon_1 - \varepsilon_y}{\varepsilon_y - \varepsilon_2} = \frac{\varepsilon_x - \varepsilon_2}{\varepsilon_1 - \varepsilon_x}. \quad (18)$$

6.1.2. Equilibrium Equations. The basic element consists of a steel bar and concrete such that its equilibrium condition can be derived from the stress state as shown in Figure 4, which can be expressed as

$$\begin{cases} f_x = \sigma_{cx} + \rho_{sx}\sigma_{sx}, \\ f_y = \sigma_{cy} + \rho_{sy}\sigma_{sy}, \\ v_{xy} = \tau_{cxy}, \end{cases} \quad (19)$$

where σ_{cx} and σ_{cy} are the normal stresses of concrete in the *x* and *y* directions, respectively; τ_{cxy} is the shear stress of concrete; ρ_{sx} and ρ_{sy} denote the reinforcement ratios in the *x*

and *y* directions, respectively; σ_{sx} and σ_{sy} are the normal stresses of the steel bar in the *x* and *y* directions, respectively.

Considering the condition of Mohr' circle of stress, the normal stresses and shear stress of concrete are obtained by

$$\begin{cases} \sigma_{cx} = \dfrac{\sigma_{c1} - \tau_{cxy}}{\tan\theta}, \\[4mm] \sigma_{cy} = \sigma_{c1} - \tau_{cxy}\tan\theta, \\[4mm] \tau_{cxy} = (\sigma_{c1} - \sigma_{c2})\sin\theta\cos\theta, \end{cases} \quad (20)$$

where σ_{c1} and σ_{c2} are the principal stresses in the 1 and 2 directions.

6.1.3. Constitutive Equations. With equations (19) and (20), it is found that the stress vector of the RC element can be obtained by the stress states of concrete and steel. Therefore, the constitutive stress-strain relations of these two materials are necessary for the state determination of the element. Especially, the steel bars are assumed in uniaxial stress state and the concrete is subjected to biaxial stress state, which can be described in the two principal directions.

For reinforcement steel, the uniaxial elastic perfectly-plastic model is adopted, which is given by

$$\begin{cases} \sigma_{sx} = E_{sx}\varepsilon_{sx} \le f_{yx}, \\ \sigma_{sy} = E_{sy}\varepsilon_{sy} \le f_{yy}, \end{cases} \quad (21)$$

where E_{sx}, ε_{sx}, and f_{yx} are the elastic modulus, strain, and yielding strength of the steel bar in the *x* direction, respectively; E_{sy}, ε_{sy}, and f_{yy} are the elastic modulus, strain, and yielding strength of the steel bar in the *y* direction, respectively.

For concrete, the shear stress state is distinctly different from the uniaxial stress state. In consideration of the tensile stress perpendicular to the principal compressive direction having influences on the compressive behavior of concrete, it is recommended using the modified uniaxial stress-strain relationships to represent the stress-strain relationship of the RC plane element subjected to combined stress state, which are

Stress-strain relationship in the tensile principal direction

$$\sigma_{c1} = \begin{cases} E_c\varepsilon_1, & \varepsilon_1 \le \varepsilon_{cr}, \\[4mm] \dfrac{f_t}{1 + \sqrt{200\varepsilon_1}}, & \varepsilon_1 > \varepsilon_{cr}. \end{cases} \quad (22)$$

Stress-strain relationship in the compressive principal direction

TABLE 5: Comparisons of predictive performance between GBRT and general ML models.

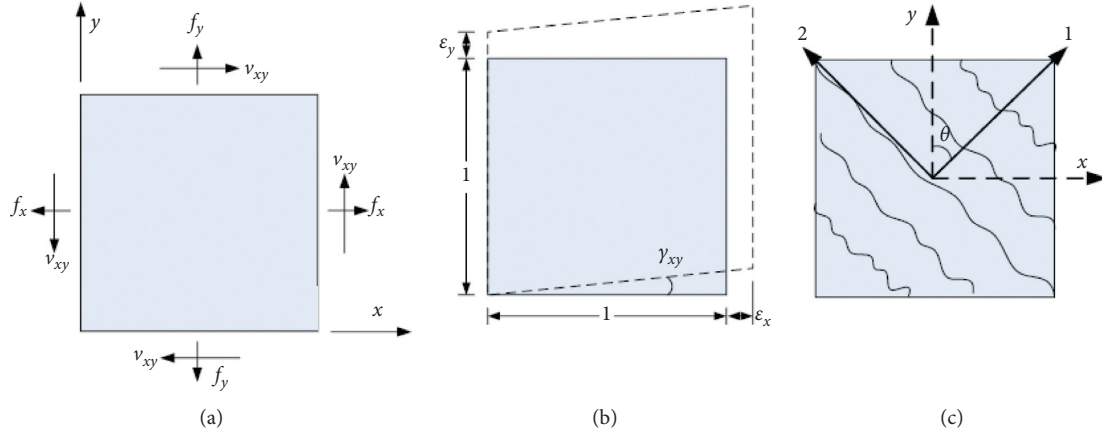| Model	R^2	RMSE (MPa)	MAE (MPa)
GBRT	0.875	0.948	0.722
LR	0.747	1.443	1.038
SVM	0.798	1.257	0.813
ANN	0.626	1.729	1.019
DT	0.802	1.285	0.840

FIGURE 4: The basic RC element state. (a) Stress state. (b) Strain state. (c) Principal direction.

$$\sigma_{c2} = f_{c2\,\max} \left[2\left(\frac{\varepsilon_2}{\varepsilon_0}\right) - \left(\frac{\varepsilon_2}{\varepsilon_0}\right)^2 \right], \tag{23}$$

with

$$\frac{f_{c2\,\max}}{f_c'} = \frac{1}{0.8 - 0.34\,(\varepsilon_1/\varepsilon_0)} \leq 1, \tag{24}$$

where E_c is the elastic modulus of concrete; f_t and f_c' are the tensile and compressive strengths of concrete, respectively; ε_{cr} and ε_0 are the strains corresponding to the tensile strength and the compressive strength, respectively; $f_{c2\,\max}$ is the maximum compressive stress in the principal compressive direction. It is clear that the modification equation (24) considers the reduction of concrete compressive strength due to the existence of tensile stress.

6.1.4. Crack Check.

Note that the abovementioned equations handle with the global behavior of the element in an average sense, while it cannot provide the local behavior description. The local equilibrium across a crack should also be satisfied, say,

$$\begin{cases} \rho_{sx}\left(\sigma_{sxcr} - \sigma_{sx}\right) = \dfrac{\sigma_{c1} + f_{ci} + v_{ci}}{\tan\theta}, \\[2mm] \rho_{sy}\left(\sigma_{sycr} - \sigma_{sy}\right) = \sigma_{c1} + f_{ci} + v_{ci}\tan\theta, \end{cases} \tag{25}$$

where σ_{sxcr} and σ_{sycr} are the steel stress at the crack; f_{ci} and v_{ci} are the local compressive and shear stresses at the crack,

respectively. The abovementioned equation can be satisfied if there are no local compressive and shear stresses, say,

$$\rho_{sx}\left(\sigma_{sxcr} - \sigma_{sx}\right) = \rho_{sy}\left(\sigma_{sycr} - \sigma_{sy}\right) = \sigma_{c1}. \tag{26}$$

However, a constrain should be ensured that the steel stresses at the crack should not exceed the yield strength of the steel, i.e., $(\sigma_{sxcr}/\sigma_{sycr}) < (f_{yx}/f_{yy})$. Therefore, if this condition is not satisfied, the local stresses should be calculated iteratively. The expressions for the local stresses are

$$v_{ci} = 0.18v_{ci,\max} + 1.64 f_{ci} - 0.82\frac{f_{ci}^2}{v_{ci,\max}},$$

$$v_{ci,\max} = \frac{\sqrt{-f_c}}{(0.31 + 24w)/(a + 16)}, \tag{27}$$

where w is the crack width; a is the maximum aggregate size; f_{ci} is calculated according to ref [4].

The whole process of using MCFT applied to the shear strength of internal RC joints can be depicted in Figure 5. More details can also be found in [4].

6.2. Comparison between GBRT and MCFT.

To further evaluate the performance of the GBRT model, the conventional MCFT is also used to predict the shear strength of the 86 RC internal beam-column joints. The statistic results from the MCFT model are compared with the GBRT model and shown in Table 6. Note that to fairly compare the performance of the two models, the prediction results in the 10 testing sets of the 10-fold cross-validation process are used for the GBRT model.
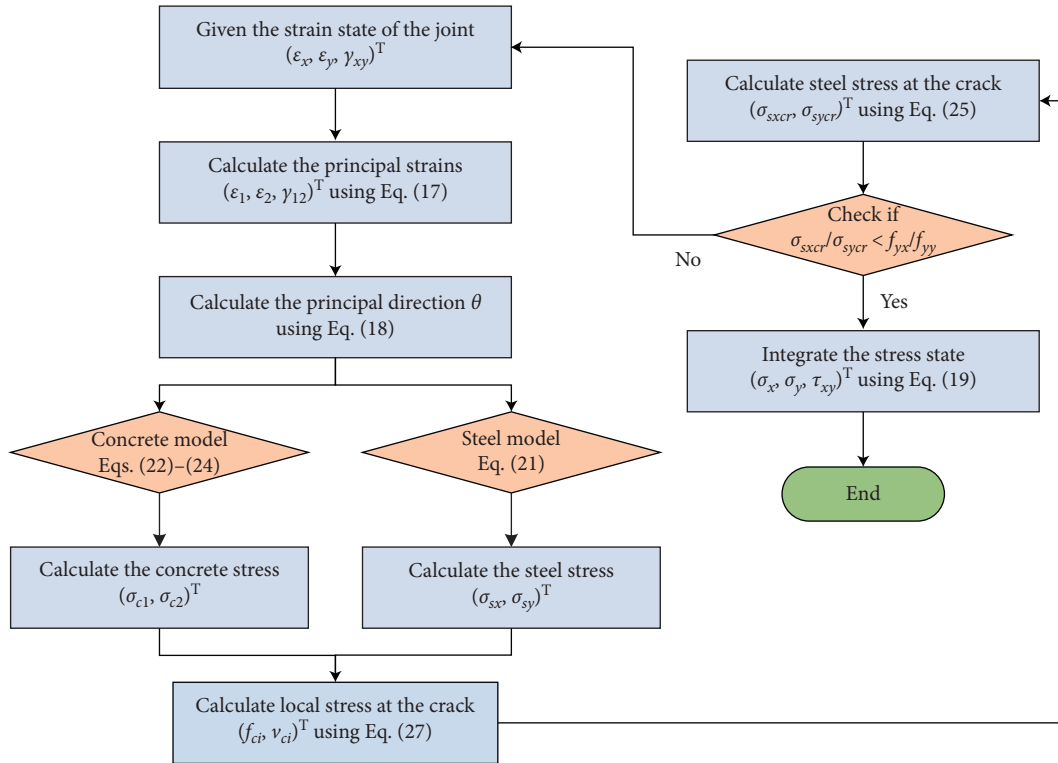
Figure 5: Flowchart of implementation of MCFT in calculating shear strength of internal RC joints.

Table 6: Comparisons of predictive performance between GBRT and MCFT models.

Model	R^2	RMSE (MPa)	MAE (MPa)
MCFT	0.765	1.355	1.066
GBRT	0.960	0.562	0.380
(GBRT-MCFT)/MCFT (%)	25.4	−58.5	−64.3

As can be seen from Table 6, the determination coefficient of the GBRT model has been improved by 25.4% and closer to 1 compared to the mechanical-driven MCFT model, and all the other two indicators have been dropped more than 50%. In other words, the ML-based method has obviously better performance than the MCFT-based method. Furthermore, the predicted and tested values are also plotted in Figure 6. Evidently, the GBRT results match the experimental results much better than those of the MCFT model.

Table 7 gives the statistic results of predicted value/tested value ratios for the MCFT and GBRT models. It can be concluded from Table 7 that the GBRT model statistically underestimates the shear strength because the mean value is less than 1, while the MCFT model slightly overestimates the shear strength. Apparently, the mean predicted value/tested value ratios for the GBRT approach is closer to 1 with less dispersion (St.D.) compared to the MCFT method.

Figure 7 further illustrates the predicted value/tested value ratios for the GBRT and MCFT models. The solid line, the top dotted line, and the bottom line represent the mean value, mean value plus St.D., and mean value minus St.D., respectively. Evidently, better prediction performance is achieved by the GBRT model.

7. Model Sensitivity Analysis

7.1. Sensitivity of Input Parameters. With the developed GBRT model, it is convenient for us to investigate the influences of different parameters on the shear strength of the internal RC joint and even quantify the influences. In this study, 10 input variables with different value ranges are adopted to conduct a comprehensive parametric analysis. In the parametric analysis, the control variable method is used, i.e., one control parameter varies, while other parameters are fixed. The specimen J6 of [44] is used as the reference model. The numerical ranges of the 10 inputs are shown in Table 8.

Figure 8 shows the predicted shear strength of the internal RC joints with different input variables by using the GBRT model. It can be drawn from Figure 8 that among all the input variables, concrete strength f_c is the most significant parameter affecting the shear strength. With the increasing of concrete strength f_c, beam width b_b, column width b_c, column height h_c, yielding strength of column longitudinal bar $f_{y,c}$, yielding strength of joint transverse bar f_{yv}, transverse bar ratio ρ_v, or axial load ratio n, the shear strength has a general ascending trend. On the contrary, yielding strength of beam longitudinal
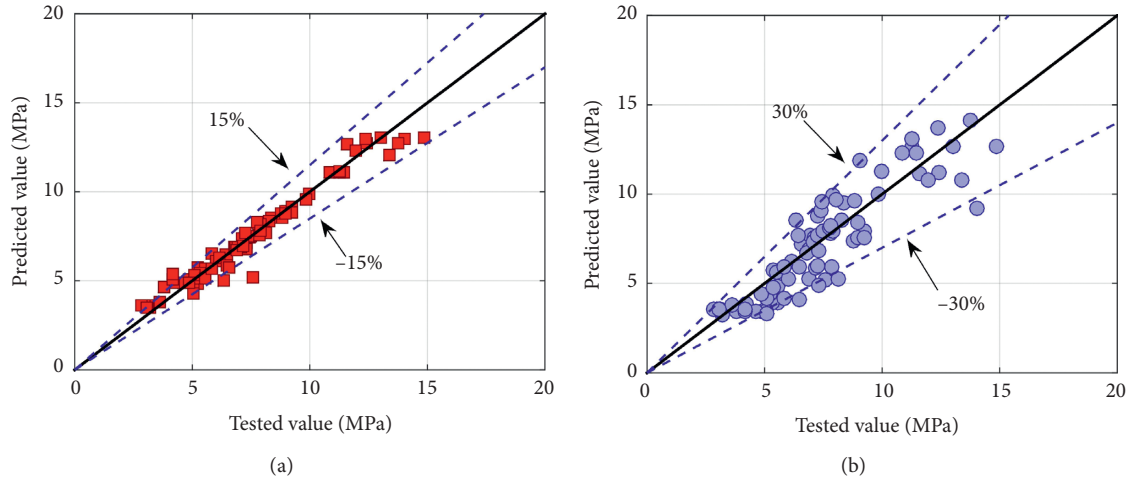
FIGURE 6: Comparisons of shear strength of RC joints by the GBRT predictive model and MCFT model. (a) GBRT results. (b) MCFT results.

TABLE 7: Statistic results of predicted value/tested value ratios for the MCFT and GBRT models.

Model	Mean value	St.D.
MCFT	0.961	0.180
GBRT	1.012	0.092
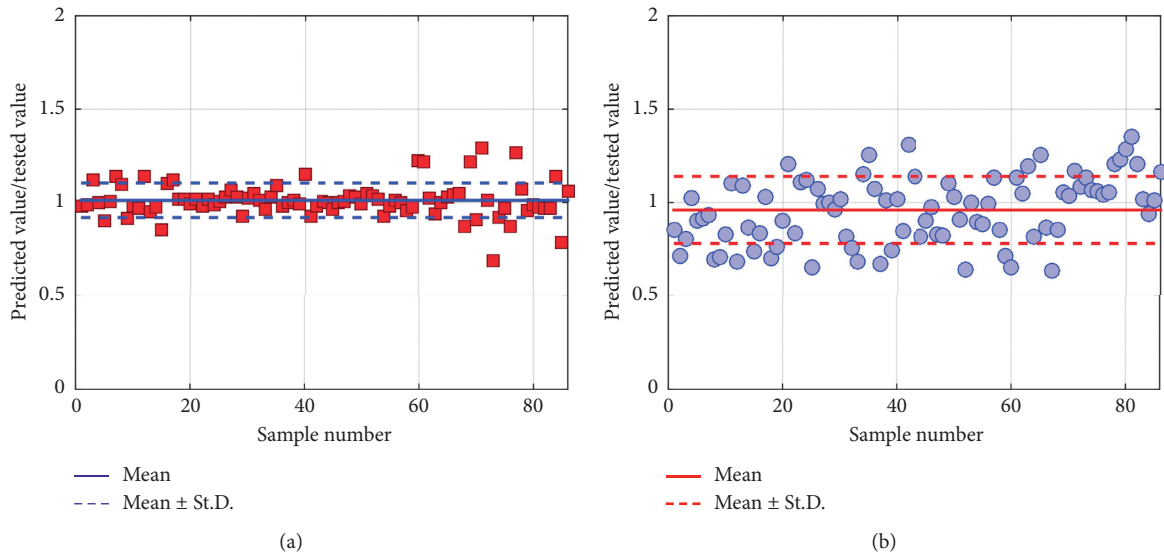
FIGURE 7: Comparisons of the predicted value/tested value for the GBRT and MCFT models. (a) GBRT results. (b) MCFT results.

TABLE 8: Numerical ranges of input variables.

f_c (MPa)	b_c (mm)	h_c (mm)	b_b (mm)	h_b (mm)
[20 : 20 : 100]	[200 : 100 : 500]	[200 : 100 : 500]	[150 : 100 : 450]	[200 : 100 : 500]
$f_{y,b}$ (MPa)	$f_{y,c}$ (MPa)	f_{yv} (MPa)	ρ_v (%)	n
[300 : 140 : 1000]	[300 : 140 : 1000]	[300 : 140 : 1000]	[0.2 : 0.2 : 1.0]	[0 : 0.1 : 0.6]

bar $f_{y,b}$ has negative effects on the shear strength. The influence of beam height h_b on the shear strength is negligible.

7.2. *Feature Importance.* Feature importance, which is used to quantify the importance of the input variables (or features), is conducted to further investigate the sensitivity of

(a)

(b)

(c)

(d)

(e)

(f)
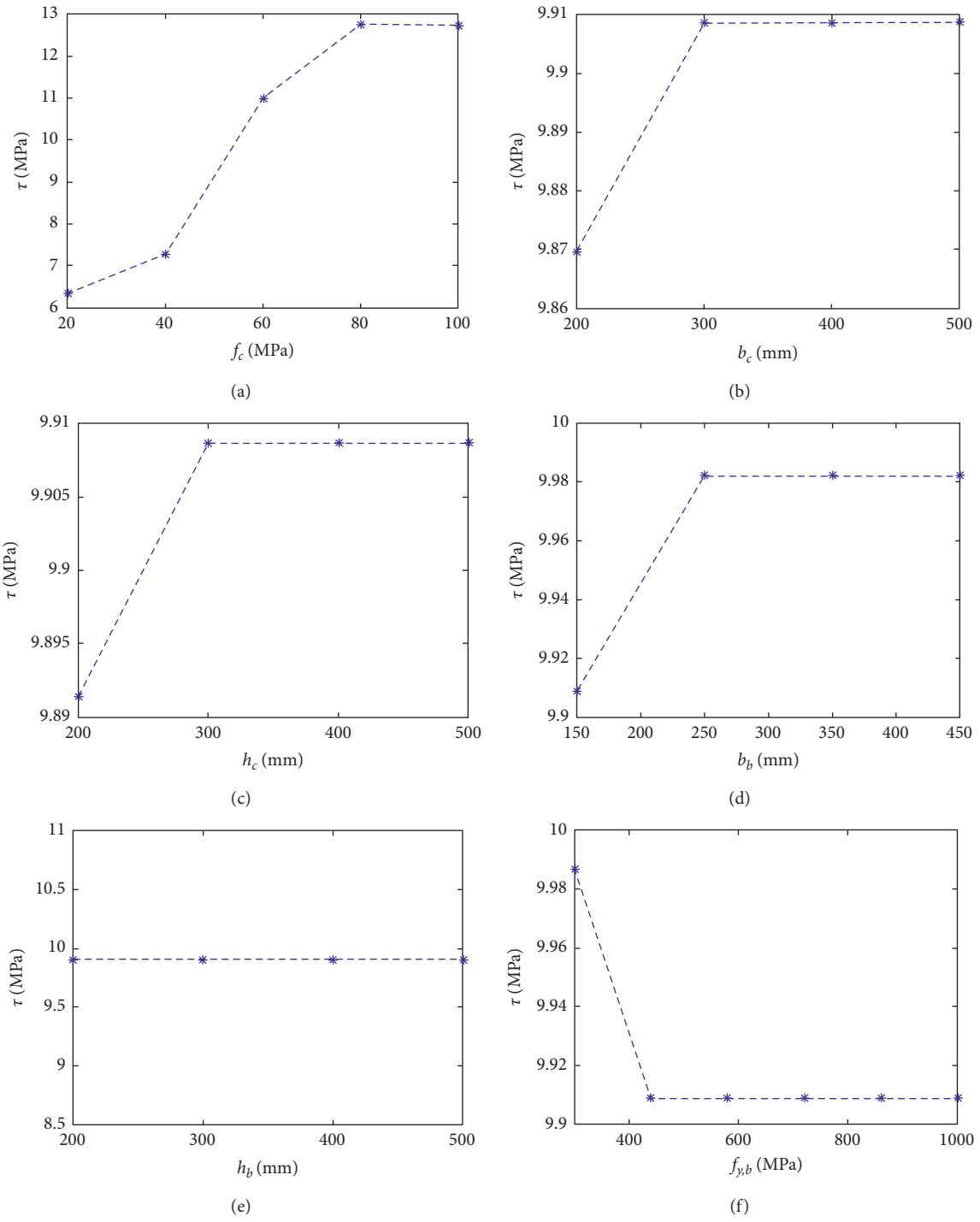
Figure 8: Continued.

(g)

(h)

(i)

(j)
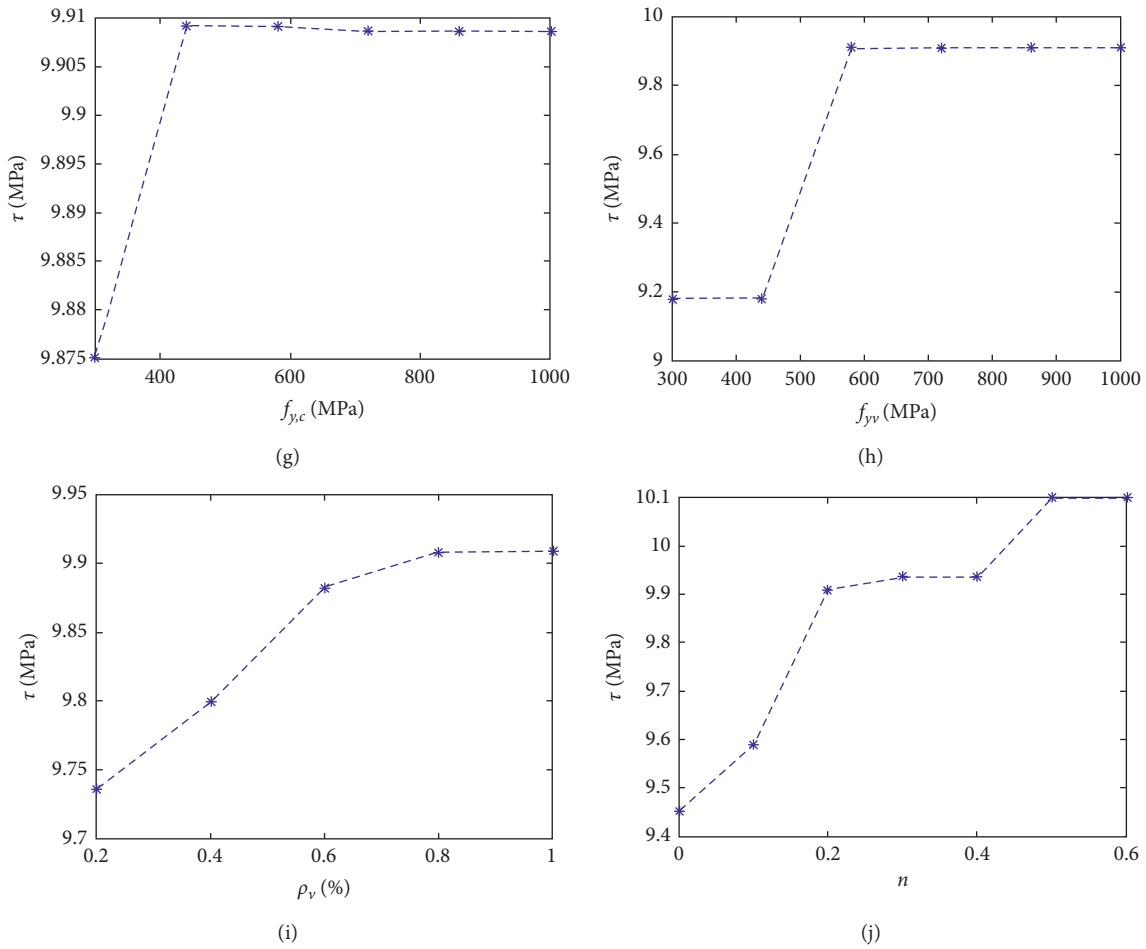
Figure 8: Prediction results of parametric analysis. (a) Concrete strength (MPa). (b) Section width of column (mm). (c) Section height of column (mm). (d) Section width of beam (mm). (e) Section height of beam (mm). (f) Yielding strength of beam longitudinal bar (MPa). (g) Yielding strength of column longitudinal bar (MPa). (h) Yielding strength of joint transverse bar (MPa). (i) Transverse bar ratio (%). (j) Axial load ratio.

each input variable on the shear strength of the internal RC joints. The calculation of feature importance can be summarized as follows. Firstly, some out-of-bag (OOB) samples are selected. Secondly, the values of the target input variable are randomly shuffled while other inputs remain unchanged. Then, the feature importance can be calculated as the accuracy difference of the two predictions using the GBRT model. Figure 9 shows the relative feature importance of all input variables. It is clear that concrete strength f_c is the key feature determining the shear strength of the internal RC joints, which is in accordance with the conclusion obtained from the previous subsection. The influences of the yielding strength of joint transverse bar f_{yv}, transverse bar ratio ρ_v, and axial load ratio n on shear strength are subdominant. The remaining input variables are insignificant features. The feature importance results are also in accordance with the sensitivity results performed before.
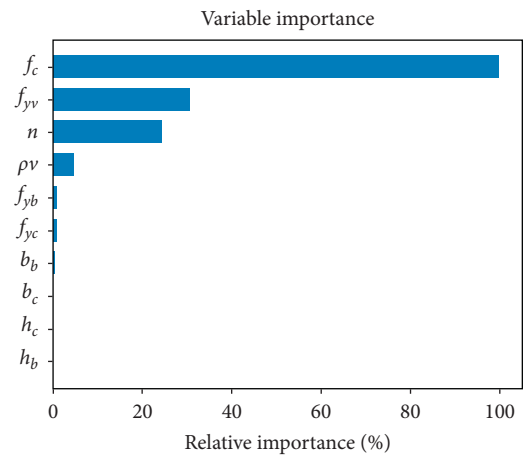
Figure 9: Relative feature importance of all input variables.

8. Conclusions

This paper presents a ML-based approach to predict the shear strength of internal RC beam-column joints. One of the famous ensemble learning methods, GBRT, is employed to solve the prediction problem. A database of 86 sets of internal RC joint tests is collected from the literature. Some individual-type ML methods and the conventional MCFT method are adopted for comparisons of the developed GBRT prediction model. The model sensitivity analysis of input parameters is conducted for the proposed GBRT-based model. Based on the results, the following conclusions can be drawn:

(1) The GBRT model can accurately and efficiently predict the shear strength of internal RC beam-column joints with given input variables.

(2) If 80% of the whole dataset is used to train the GBRT model, the average determination coefficient R^2 of the 10-fold cross-validation is 0.875, which means that the prediction error is low. Meanwhile, the average RMSE and MAE are 0.948 MPa and 0.722 MPa, indicating that the prediction model has a low prediction deviation.

(3) Among all the ML-based prediction models used in this study, the GBRT model performs best with the closest R^2 to 1 and smallest values of RMSE and MAE. It indicates that the GBRT model is superior to the individual-type ML algorithms.

(4) The GBRT model has better prediction performance compared with the conventional MCFT model in both average sense and variance sense and exhibits a significant superiority in terms of the three performance indicators.

(5) Among all the input variables, concrete strength f_c is the most critical feature affecting the shear strength of the internal RC joints. With the increasing of the concrete strength, the shear strength significantly increases. Other input variables are relatively subordinate or even unimportant.

Data Availability

The data will be available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

References

[1] N. W. Hanson and H. W. Connor, "Seismic resistance of reinforced concrete beam–column joints," *Journal of Structural Division ASCE*, vol. 11, no. 56, pp. 533–559, 1967.

[2] H. Dabiri, A. Kheyroddin, and A. Kaviani, "A numerical study on the seismic response of RC wide column-beam joints," *International Journal of Civil Engineering*, vol. 17, no. 3, pp. 377–395, 2019.

[3] R. Costa, P. Providência, and A. Dias, "Component-based reinforced concrete beam-column joint model," *Structural Concrete*, vol. 18, no. 1, pp. 164–176, 2017.

[4] F. J. Vecchio and M. P. Collins, "The modified compression-field theory for reinforced concrete elements subjected to shear," *ACI Structural Journal*, vol. 83, no. 2, pp. 219–231, 1986.

[5] T. Paulay and M. J. N. Priestley, *Seismic Design of Reinforced Concrete and Masonry Building*, Wiley, New York, NY, USA, 1992.

[6] X. Long and C. K. Lee, "Improved strut-and-tie method for 2D RC beam-column joints under monotonic loading," *Computers and Concrete*, vol. 15, no. 5, pp. 807–831, 2015.

[7] C. Lima, E. Martinelli, and C. Faella, "Capacity models for shear strength of exterior joints in RC frames: state-of-the-art and synoptic examination," *Bulletin of Earthquake Engineering*, vol. 10, no. 3, pp. 967–983, 2012.

[8] T. S. Eom, H. J. Hwang, and H. G. Park, "Energy-based hysteresis model for reinforced concrete beam-column connections," *ACI Structural Journal*, vol. 112, no. 2, pp. 157–166, 2015.

[9] ASCE/SEI 41, *Seismic Rehabilitation of Existing Buildings*, American Society of Civil Engineers, Reston, VA, USA, 2007.

[10] H.-J. Hwang, T.-S. Eom, and H.-G. Park, "Shear strength degradation model for performance-based design of interior beam-column joints," *ACI Structural Journal*, vol. 114, no. 5, pp. 1143–1154, 2017.

[11] H. J. Hwang and H. G. Park, "Requirements of shear strength and hoops for performance-based design of interior beam-column joints," *ACI Structural Journal*, vol. 116, no. 2, pp. 245–256, 2019.

[12] H.-J. Hwang and H.-G. Park, "Performance-based shear design of exterior beam-column joints with standard hooked bars," *ACI Structural Journal*, vol. 117, no. 2, pp. 67–80, 2020.

[13] S.-J. Hwang, R.-J. Tsai, W.-K. Lam, and J. P. Moehle, "Simplification of softened strut-and-tie model for strength prediction of discontinuity regions," *ACI Structural Journal*, vol. 114, no. 5, pp. 1239–1248, 2017.

[14] R. Y. C. Huang and J. S. Kuang, "Predicting strength of exterior wide beam-column joints for seismic resistance," *Journal of Structural Engineering*, vol. 146, no. 2, Article ID 04019209, 2020.

[15] W. M. Hassan and J. P. Moehle, "Shear strength of exterior and corner beam-column joints without transverse reinforcement," *ACI Structural Journal*, vol. 115, no. 6, pp. 1719–1727, 2018.

[16] ACI Committee 318, *Building Code Requirements for Structural Concrete (ACI 318-14) and Commentary (ACI 318R-14)*, American Concrete Institute, Farmington Hills, MI, USA, 2014.

[17] S. Akkurt, G. Tayfur, and S. Can, "Fuzzy logic model for the prediction of cement compressive strength," *Cement and Concrete Research*, vol. 34, no. 8, pp. 1429–1433, 2004.

[18] A. Baykasoğlu, T. Dereli, and S. Tanış, "Prediction of cement strength using soft computing techniques," *Cement and Concrete Research*, vol. 34, no. 11, pp. 2083–2090, 2004.

[19] G. Trtnik, F. Kavčič, and G. Turk, "Prediction of concrete strength using ultrasonic pulse velocity and artificial neural networks," *Ultrasonics*, vol. 49, no. 1, pp. 53–60, 2009.

[20] R. Siddique, P. Aggarwal, and Y. Aggarwal, "Prediction of compressive strength of self-compacting concrete containing bottom ash using artificial neural networks," *Advances in Engineering Software*, vol. 42, no. 10, pp. 780–786, 2011.

[21] A.-D. Pham, N.-D. Hoang, and Q.-T. Nguyen, "Predicting compressive strength of high performance concrete using metaheuristic-optimized least squares support vector regression," *Journal of Computing in Civil Engineering*, vol. 30, no. 3, Article ID 06015002, 2015.

[22] B. A. Omran, Q. Chen, and R. Jin, "Comparison of data mining techniques for predicting compressive strength of environmentally friendly concrete," *Journal of Computing in Civil Engineering*, vol. 30, no. 6, Article ID 04016029, 2016.

[23] P. G. Asteris and V. G. Mokos, "Concrete compressive strength using artificial neural networks," *Neural Computing and Applications*, vol. 32, no. 15, pp. 11807–11826, 2019.

[24] V. Plevris and P. G. Asteris, "Modeling of masonry failure surface under biaxial compressive stress using Neural Networks," *Construction and Building Materials*, vol. 55, pp. 447–461, 2014.

[25] P. G. Asteris and V. Plevris, "Anisotropic masonry failure criterion using artificial neural networks," *Neural Computing and Applications*, vol. 28, no. 8, pp. 2207–2229, 2017.

[26] J.-S. Jeon, A. Shafieezadeh, and R. DesRoches, "Statistical models for shear strength of RC beam-column joints using machine-learning techniques," *Earthquake Engineering & Structural Dynamics*, vol. 43, no. 14, pp. 2075–2095, 2014.

[27] D.-T. Vu and N.-D. Hoang, "Punching shear capacity estimation of FRP-reinforced concrete slabs using a hybrid machine learning approach," *Structure and Infrastructure Engineering*, vol. 12, no. 9, pp. 1153–1161, 2016.

[28] W. Zheng and F. Qian, "Promptly assessing probability of barge-bridge collision damage of piers through probabilistic-based classification of machine learning," *Journal of Civil Structural Health Monitoring*, vol. 7, no. 1, pp. 57–78, 2017.

[29] A. Santos, E. Figueiredo, M. Silva, R. Santos, C. Sales, and J. C. Costa, "Genetic-based EM algorithm to improve the robustness of Gaussian mixture models for damage detection in bridges," *Structural Control and Health Monitoring*, vol. 24, no. 3, p. e1886, 2017.

[30] H. Salehi and R. Burgueño, "Emerging artificial intelligence methods in structural engineering," *Engineering Structures*, vol. 171, pp. 170–189, 2018.

[31] R. J. Schalkoff, *Artificial Neural Networks*, McGraw-Hill, New York, NY, USA, 1997.

[32] M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt, and B. Scholkopf, "Support vector machines," *IEEE Intelligent Systems and Their Applications*, vol. 13, no. 4, pp. 18–28, 1998.

[33] S. R. Safavian and D. Landgrebe, "A survey of decision tree classifier methodology," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 21, no. 3, pp. 660–674, 1991.

[34] Z.-H. Zhou, "Ensemble learning," *Encyclopedia of Biometrics*, S. Z. Li, Ed., pp. 411–416, Springer Press, New York, NY, USA, 2015.

[35] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*, MIT Press, Cambridge, MA, USA, 2012.

[36] V. N. Vapnik, "An overview of statistical learning theory," *IEEE Transactions on Neural Networks*, vol. 10, no. 5, pp. 988–999, 1999.

[37] Research Group on Frame Joints, "Shear strength of reinforced concrete beam-column joints under low reversed cyclic loading," *Journal of Building Structures*, vol. 4, no. 6, pp. 1–17, 1983, in Chinese.

[38] C. Zhao, D. Zhang, T. Wang, and D. Chen, "Experimental study on the seismic property of the beam-column joints in high strength concrete frame under alternating load," *Journal of Shenyang Architectural and Civil Engineering Institute*, vol. 9, no. 3, pp. 260–268, 1993, in Chinese.

[39] X. Lu, Z. Guo, and Y. Wang, "Experimental study on seismic behavior of beam-column sub-assemblages in RC frame," *Journal of Building Structures*, vol. 22, no. 1, pp. 2–7, 2001, in Chinese.

[40] Q. Yu and S. Li, "Research on frame's joint that concrete strength of core is inferior to that of column," *Journal of Tongji University*, vol. 32, no. 12, pp. 1583–1588, 2004, in Chinese.

[41] B. Xu, M. Cheng, M. Zhang, and J. Qian, "Experimental study on behavior of reinforced concrete beam-column joint with lower core concrete strength," *Building Structures*, vol. 36, no. 6, pp. 18–23, 2006, in Chinese.

[42] S. Otani, K. Kitayama, and H. Aoyama, "Reinforced concrete interior beam-column joints under simulated earthquake loading," in *Proceedings of the US-New Zealand-Japan Seminar on Design of Reinforced Concrete Beam-Column Joints*, Monterey, Canada, 1984.

[43] D. E. Meinheit and J. O. Jirsa, "The shear strength of reinforced concrete beam-column joints," CESRL Report No. 77-l, University of Texas, Austin, TX, USA, 1977.

[44] H. Noguchi and T. Kashiwazaki, "Test on high-strength concrete interior beam-column joints," in *Proceedings of the Tenth World Conference on Earthquake Engineering*, Madrid, Spain, 1992.

[45] K. Oka and H. Shiohara, "Test on high strength concrete interior beam-column sub-assemblage," in *Proceedings of The Tenth World Conference on Earthquake Engineering*, Madrid, Spain, 1992.

[46] Y. Higashi and Y. Ohwada, "Failing behaviors of reinforced concrete beam-column connections subjected to lateral load," *Memories of Faculty of Technology Tokyo Metropolitan University*, vol. 19, pp. 91–101, University of Virginia, Charlottesville, VA, USA, 1969.

[47] M. Teraoka, Y. Kanoh, K. Tanaka, and I. C. Hayashi, "Strength and deformation behavior of RC interior beam-and-column joints using high strength concrete," in *Proceedings of the Second US-Japan-New Zealand-Canada Multilateral Meeting on Structural Performance of High Strength Concrete in Seismic Regions*, Honolulu, HI, USA, 1994.

[48] S. S. Zaid, "Behavior of reinforced concrete beam-column connections under earthquake loading," Doctoral dissertation, University of Tokyo, Tokyo, Japan, 2001.

[49] S. Fujii and S. Morita, "Comparison between interior and exterior RC beam-column joint behavior," *Design of Beam-Column Joints for Seismic Resistance, SP-123*, pp. 145–165, ACI, Detroit, MI, USA, 1991.

[50] D. E. Abrams, "Scale relations for reinforced concrete beam-column joints," *ACI Structural Journal*, vol. 84, no. 6, pp. 502–512, 1987.

[51] R. T. Leon, "Shear strength and hysteretic behavior of interior beam-column joints," *ACI Structural Journal*, vol. 87, no. 1, pp. 3–11, 1990.

[52] T. Wu, X. Liu, and H. Wei, "Experimental study on seismic behavior of frame interior joints with high-strength lightweight aggregate reinforced concrete," *China Civil Engineering Journal*, vol. 51, no. 6, pp. 32–42, 2018, in Chinese.

[53] H. H. Zhang, Z. L. Lu, and S. Z. Su, "Experimental study on seismic behavior of high-strength ceramic concrete frame joints," *Building Science*, vol. 32, no. 1, pp. 81–87, 2016, in Chinese.

[54] B. Qu, "Experimental study on seismic performance of full lightweight aggregate concrete beam-column joints," *Master's thesis*, Jilin Jianzhu University, Changchun, China, 2015, in Chinese.

[55] C. L. Decker, M. A. Issa, and K. F. Meyer, "Seismic investigation of interior reinforced concrete sand-lightweight concrete beam-column joints," *ACI Structural Journal*, vol. 112, no. 3, pp. 287–297, 2015.

Research Article

Application of Residual Shear Strength Predicted by Artificial Neural Network Model for Evaluating Liquefaction-Induced Lateral Spreading

Yanxin Yang [iD],[1] **Bai Yang** [iD],[1,2] **Chunhui Su**,[1] **and Jianlin Ma**[2]

[1]*School of Architecture and Transportation Engineering, Guilin University of Electronic Technology, Guilin, Guangxi 541004, China*
[2]*School of Civil Engineering, Southwest Jiaotong University, Chengdu, Sichuan 610031, China*

Correspondence should be addressed to Bai Yang; ayangbai@163.com

The residual shear strength of liquefied soil is critical to estimating the displacement of lateral spreading. In the paper, an Artificial Neural Network model was trained to predict the residual shear strength ratio based on the case histories of lateral spreading. High-quality case histories were analyzed with Newmark sliding block method. The Artificial Neural Network model was used to predict the residual shear strength of liquefied soil, and the post-liquefaction yield acceleration corresponding with the residual shear strength was obtained by conducting limit equilibrium analysis. Comparing the predicted residual shear strength ratios to the recorded values for different case histories, the correlation coefficient, *R*, was 0.92 and the mean squared error (MSE) was 0.001 for the predictions by the Artificial Neural Network model. Comparison between the predicted and reported lateral spreading for each high-quality case history was made. The results showed that the probability of the lateral spreading calculated with the Newmark sliding block method using the residual shear strength was 98% if a lateral spreading ratio of 2.0 was expected and a truncated distribution was used. An exponential relationship was proposed to correlate the residual shear strength ratio to the equivalent clean sand corrected SPT blow count of the liquefied soil.

1. Introduction

Liquefaction is the phenomenon whereby saturated sandy soil behaves like a liquid during the shaking by earthquakes. In the saturated sandy soil, the strength of the soil is lost when the pore water pressure builds up and approaches the total pressure due to the cyclic shear loading. During the past earthquakes, widespread damage caused by liquefaction includes the damage to buildings, pipelines, coastal slopes, and the ground deformations.

The limited deformation of gently sloping ground is defined as lateral spreading. To adopt proper mitigation measures, the displacement magnitude of lateral spreading needs to be determined. The residual shear strength, which corresponds to the minimum shear strength at which the lateral spreading is mobilized, is one of the main factors to determine the lateral spreading induced by seismic liquefaction.

Several methods have been proposed to estimate the residual strength of liquefied soil based on the back-calculation of the case histories or laboratory tests. Compared to the laboratory test, the case histories make it possible to consider the effect of voids redistribution, inertial effect, geometry effect, and the other factors on the residual shear strength of liquefied soil, so the method estimating the residual shear strength back-calculated from the case histories is widely used in the seismic displacement calculation and stability analysis of the level or sloping ground.

There are disadvantages when utilizing the residual shear strength relationship developed from case histories. As the development of the estimation methods is based on in-site investigation and the soil resistance parameters (Standard

Penetration Test or Cone Penetration Test values of the liquefiable soil) of case histories, the limited number of the case histories will result in the uncertainties in estimating the residual shear strength. On the other hand, the back-calculation of residual shear strength is depending on the case histories of flow failures, initially proposed by Seed [1] and then modified by the other researchers. The displacement of flow failures can be more than several meters as the soil above the liquefied soil is still moving even after the earthquake stops. The postliquefaction residual shear strength of the liquefied soil is smaller than the driving force provided by the static stress. Regarding the lateral spreading, the postliquefaction residual shear strength is usually greater than the static driving force, so the movement of the soil above the liquefied soil will cease once the earthquake stops. It may not be appropriate to use the residual shear strength of liquefied soil derived from the case histories of flow failures to calculate the residual shear strength of liquefied soil at the lateral spreading sites. Although Olson and Johnson [2] stated that "lateral spreads back analyzed using the Newmark sliding block procedure exhibit mobilized strength ratios essentially identical to liquefied strength ratios back calculated from flow failures," the residual shear strength of liquefied soil in the lateral spreading cases needs to be evaluated carefully for assessing the deformation of lateral spreading in different analyses such as limit equilibrium analysis, finite difference analysis, or finite element analysis.

Recently, the Artificial Intelligence techniques, such as Artificial Neural Network [3], Support Vector Machine [4], and Genetic Programming [5], have been used in different fields of engineering. Among these Artificial Intelligence techniques, Artificial Neural Network is a powerful tool to describe the relationship between the residual shear strength and soil resistance parameters. By training the data given to the training system, the Artificial Neural Network will identify the relationship between the input and output variables.

When the residual shear strength of liquefied soil, which is corresponding with the minimum strength that the lateral spreading is mobilized, is determined, a postliquefaction yield acceleration using the residual shear strength could be obtained. As the lateral spreading is the horizontal displacement of soil underlain by liquefied soil after the triggering of liquefaction, the displacement of lateral spreading can be calculated with Newmark sliding block method. The mechanism of liquefaction-induced lateral spreading can be described as follows: The downslope displacement accumulates when the seismic force is oriented to downslope direction, and the liquefied soil retains the residual shear strength during the accumulation of the displacement. The failure is driven by the combined static shear stress and the seismic force, which is greater than the residual shear strength, while the static stress is less than the residual shear strength, so the displacement of lateral spreading ceases when the earthquake ends. Such an accumulation of displacement can be calculated by the Newmark sliding block method if the soil above the liquefied soil is treated as a rigid block and it slides over the sliding surface existing in the

liquefied soil. Despite that the application of the Newmark sliding block method in lateral spreading has been proposed and conducted by Baziar et al. [6], Taboada et al. [7], and Kavazanjian [8], systematic research on its application by analyzing more case histories of lateral spreading is needed.

In this paper, firstly, the Artificial Neural Network model was trained to predict the residual shear strength ratio (the ratio of residual shear strength to the effective shear stress) in terms of equivalent clean sand corrected SPT value of the liquefiable soil. A database of lateral spreading was used for training the neurons to recognize the patterns between the residual shear strength ratio and equivalent clean sand corrected SPT blow count of the liquefied sand. The statistical performance of the model was evaluated based on the two parameters including correlation coefficient, R, and mean square error (MSE). The cumulative probabilities were calculated for different ratios (i.e., the ratio of the predicted residual shear strength to the observed residual shear strength). By computing the root mean square error (RMSE), the Artificial Neural Network model was compared with the Olson and Johnson model [2]. A high-quality database of lateral spreading was established based on the case histories used in the process of training the Artificial Neural Network model. The residual shear strength predicted by the Artificial Neural Network model for high-quality case histories of lateral spreading was subsequently analyzed in the limit equilibrium analysis [9] to obtain postliquefaction yield acceleration. The Newmark sliding block analysis implemented in a computer code (SLAMMER) [10] was conducted for high-quality case histories. By applying a truncated normal distribution to describe the distribution of lateral spreading ratio (the ratio of predicted lateral spreading to observed lateral spreading), a statistical analysis was conducted to evaluate the accuracy of the Newmark sliding block method. Based on the residual shear strength values of the high-quality case histories, an exponential equation was proposed to represent the relationship between the residual shear strength ratio and the equivalent clean sand corrected SPT blow count of the liquefied soil.

1.1. Residual Shear Strength of Liquefied Soil.
The residual shear strength of liquefied sand could be estimated by empirical models based on the SPT (Standard Penetration Test) blow counts of the soil. Seed [1] back-analyzed limited cases of liquefaction flow failures. The available residual shear strengths of liquefied sand and equivalent clean sand $(N_1)_{60\text{-cs}}$ values were summarized, and a relationship between the residual shear strength and equivalent clean sand $(N_1)_{60\text{-cs}}$ values was proposed in a form of a chart. Seed and Harder [11] updated the chart of Seed [1], and with more case histories data put in the database, the inertial effects were taken into consideration in the back-analysis. The relationship based on field data provides an estimation of the undrained shear strength of liquefied sands by Standard Penetration Test corrected blow count, with an upper bound and a lower bound of residual shear strength for the specific $(N_1)_{60\text{-cs}}$ value. Idriss [12] proposed a relationship between the undrained residual shear strength and equivalent clean

sand corrected SPT blow count by modifying the database of Seed and Harbor [11]. The relationship was expressed in terms of a single exponential curve, the residual shear strength could be obtained from a specific SPT value, and error bars were used for indicating the uncertainty in calculating residual shear strength. Stark and Mersi [13] defined the ratio of back-calculated residual shear strength to the initial vertical effective stress as S_r/σ'_{vo} for 20 case histories and developed the ratio as a function of SPT blow count. Olson and Stark [14] updated the database with more case-history data and expressed the residual shear strength ratio as a function of equivalent clean sand corrected SPT blow count $(N_1)_{60}$ and CPT resistance q_{c1}. The fines content correction was not considered in development of the relationships. Idriss and Boulanger [15, 16] developed a relationship between the residual shear strength ratio S_r/σ'_{vo} and the equivalent clean sand corrected SPT blow count $(N_1)_{60\text{-}cs}$ using the databases of Seed [1], Seed and Harder [11], and Olson and Stark [14]. The unreliable cases were removed from the database. The relationship was expressed in a curve including two branches: one branch for the condition that the void redistribution is expected to be significant and one for the condition that void redistribution effects are expected to be negligible. Kramer and Wang [17, 18] developed a hybrid relationship that defined the ratio of residual shear strength to preearthquake vertical effective stress as a nonlinear function of normalized SPT resistance; the effective-stress-dependent approach combined classical and normalized strength approaches and was calibrated with the case histories of flow failures. Fines content correction was not suggested due to the lack of systematic variation in the residual shear strength with fines content. Olson and Johnson [2] back-analyzed a database consisting of 39 well-documented lateral spreads induced by liquefaction with Newmark sliding method. The relationship between back-calculated strength ratio S_r/σ'_{vo} and either normalized cone penetration resistance or standard penetration resistance (with no fines content correction) was developed. Olson and Johnson [2] found that the mobilized strength ratio was independent of lateral displacement magnitude and shaking intensity. Özener [19] summarized the database of lateral spreading and proposed the estimation relationship developed from the shear wave velocity and residual shear strength ratio. As a summary, the residual shear strength can be estimated by taking account of the effective overburden stress and the soil resistance parameters such as SPT blow count and cone penetration resistance. These researches suggested that the residual shear strength was related to the soil resistance parameters (such as SPT blow count or CPT resistance value) and effective overburden stress. The effect of overburden effective stress on the residual shear strength was minimized in these researches via the use of the residual shear strength ratio.

By analyzing the reported SPT blow counts and the residual shear strength ratios for the liquefied soil from the database of lateral spreading, it provides an opportunity to investigate the residual shear strength of the liquefied soil using Artificial Neural Network model. It has to be noticed that, except for the residual shear strength relationship proposed by Olson and Johnson [2] and Özener [19], the other relationships or empirical models used to calculate the residual shear strength of liquefied soil are developed based on the case histories of flow failures. Due to the difference between residual shear strength of liquefied soil for lateral spreading and flow failures, it is more suitable to use the empirical model of residual shear strength of liquefied soil developed based on the case histories of lateral spreading solely when lateral spreading is analyzed. The variances of reported residual shear strength, the effective overburden stress, and the SPT blow count are induced by instruments and different means of measurements, which further induce the uncertainties of these parameters in the research literature. In the paper, the median values of the residual shear strengths, the effective overburden stress, and the SPT blow counts are used to minimize uncertainties and obtain the best estimation of the relationship between the residual shear strength ratio (the ratio of the residual shear strength to the effective overburden stress) and the SPT blow count.

The fines content correction equation of liquefied sand by Seed [1] is used to consider the influence of fines content on the SPT blow count and the residual shear strength. The equivalent clean sand corrected SPT blow count can be obtained in equation (1), where $(N_1)_{60\text{-}cs}$ is the equivalent clean sand SPT blow count and N_{cr} is the fines content correction for the SPT blow count recommended by Seed [1], as shown in Table 1:

$$(N_1)_{60-cs} = (N_1)_{60} + N_{cr}. \qquad (1)$$

2. Introduction to Artificial Neural Network

In the field of biology, the human brain processes the information via a neural network, which is formed by billions of interconnected neurons. The Artificial Neural Network is an information system developed based on the information-processing characteristics of the human brain. In Figure 1, the structure of an Artificial Neural Network with two hidden layers is shown. The hidden pattern between the input and the output layers can be described in different forms by introducing the hidden layers. The values of the input variables are represented by the input layer, and the response of the Artificial Neural Network is represented by the output layer. The hidden layer includes the weights of input variables, the biases, and the transfer functions, responsible for training the interconnected neurons and recognizing the relationship between the input layer and the output layer. By using the transfer functions, the input weight is obtained by a trial-and-error procedure when the predicted output is approaching the observed output, and the coverages reach a local or global optimum. An error function needs to be minimized during the learning process to avoid overfitting of the variable weight. After the training is finished, the Artificial Neural Network can be tested with a set of test data.

In this paper, a two-layer feedforward neural network with sigmoid hidden neurons and linear output neurons [20] is used. A total of 39 hidden neurons are used. A Bayesian regularization method is used as a backpropagation

TABLE 1: N_{cr} for fines content.

Fines content (passing No. 200 sieve)	10%	25%	50%	75%
N_{cr}	1	2	4	5

Input layer $\in R^5$ Hidden layer $\in R^8$ Hidden layer $\in R^8$ Output layer $\in R^1$
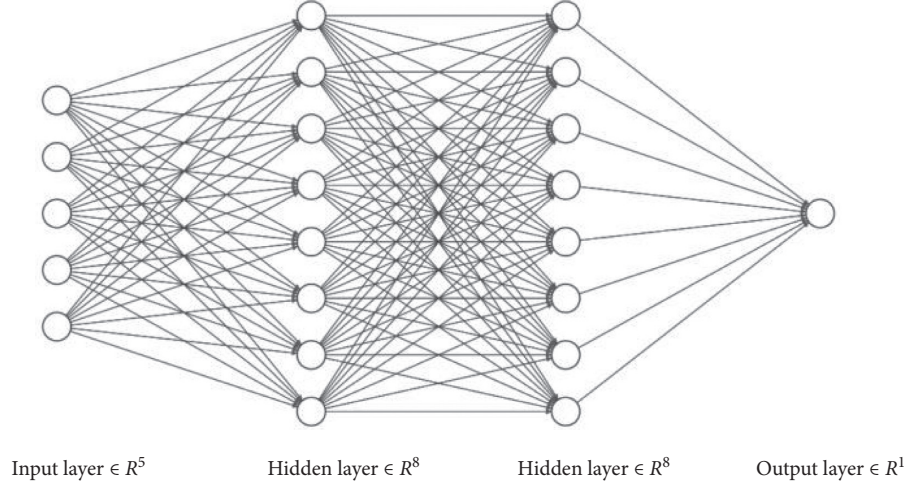
FIGURE 1: Illustration of Artificial Neural Network.

algorithm, which requires a solution to a Hessian matrix. An objective function incorporating parameters α and β and a function consisting of the mean square error of weights and biases are used to realize the regularization [21]. In equation (2), the objective function is expressed, where α and β are objective function parameters. The parameters α and β are computed using Gauss–Newton approximation of the Hessian matrix of the objective function in Bayesian framework:

$$F = \alpha E_{W} + \beta E_{D}, \qquad (2)$$

where E_{W} is the function consisting of the mean square error of weights and biases and it is expressed in equation (3), where w_i is the variable weight and M is the number of interconnected neurons:

$$E_{W} = \frac{1}{M} \sum_{i=1}^{M} w_i^2, \qquad (3)$$

where E_{D} is the sum of squared errors between the target value and the output value generated by the neural network, which is expressed in equation (4), where y_i is the target value and $\widehat{y}_i$ is the output value generated by the neural network:

$$E_{D} = \sum_{i=1}^{M} \frac{1}{2} (y_i - \widehat{y}_i)^2. \qquad (4)$$

2.1. Case Histories of Liquefied Sand in Lateral Spreading and Artificial Neural Network Results. The case histories from the different research literature, which are also reported by

Olson and Johnson [2] and Özener [19], are used to develop the database for developing and training an Artificial Neural Network model. Referring to the initial research literature for each case history, a total of 43 records were used. Among the 43 case histories, 35 records were used for training the model, 4 records were used for validation, and the rest of records were used for testing the model. The residual shear strength ratio and SPT blow count of the liquefied soil were used as input variables. In Table 2, for each case history of lateral spreading, the earthquake event, site location, residual shear strength ratio, SPT blow count of liquefied soil, fines content, median residual shear strength ratio of the liquefied soil, and the equivalent clean sand corrected SPT blow count of liquefied soil for each case history reported by the literature research are summarized. The references used for summarization are also listed in Table 2. The SPT blow count of the liquefied soil was corrected to equivalent clean sand corrected SPT blow count based on the Seed relationship [1] shown in equation (1). For equivalent clean sand corrected SPT blow count, the maximum value, minimum value, average value, and the standard deviation value are 21, 2.7, 11.1, and 3.7, respectively. The maximum value, minimum value, average value, and the standard deviation values of the residual shear strength ratios are 0.75, 0.02, 0.11, and 0.11, respectively.

In Figure 2, the observed and predicted values of residual shear strength ratios are shown to illustrate the performance of the proposed Artificial Neural Network model. The correlation coefficient, R, is 0.92 and the mean squared error (MSE) is 0.001 for the training results. In (5), the mean square error is expressed:

TABLE 2: Case histories of lateral spreading used for training Artificial Neural Network model.

Case no.	Location of lateral spreading	Earthquake event	Residual shear strength ratio	SPT blow count	Fines content (%)	Median residual shear strength ratio	Equivalent clean sand blow count	Reference
1	Heber road	1979 Imperial Valley	0.03–0.12	1.0	20	0.075	2.7	Davis et al. [22]; Castro [23]; Youd and Bennett [24]; Olson and Johnson [2]; Özener [19]
2	Wildlife site	1987 Superstition Hills	0.06–0.10	10.3	30	0.08	12.7	Holzer et al. [25]; Boulanger et al. [26]; Idriss and Boulanger [27]; Olson and Johnson [2]; Özener [19]
3	Moss Landing, MBARI Bldg. 4, Sandholdt rd. (SI-2)		0.14–0.19	10.0	5	0.17	10.5	Boulanger et al. [28]; Olson and Johnson [2]; Özener [19]
4	Moss Landing, MBARI Bldg. 4, Sandholdt rd. (SI-5)		0.11–0.19	15	<5	0.15	15.0	Boulanger et al. [28]; Olson and Johnson [2]; Özener [19]
5	Moss Landing, MLML Bldg., westward spread	1989 Loma Prieta	0.06–0.14	14.6	4	0.1	15.0	Mejia [29]; Olson and Johnson [2]; Özener [19]
6	Marina District		0.06–0.08	10.5	14–41	0.07	12.7	Olson and Johnson [2]; Özener [19]
7	Miller Farm		0.05–0.16	11.5	5–38	0.11	13.3	Holzer et al. [30]; Olson and Johnson [2]; Özener [19]
8	Treasure Island, Perimeter (T1N3)		0.07–0.22	10.0	10	0.15	11.0	Power et al. [31]; Olson and Johnson [2]; Özener [19]
9	Magsaysay Bridge E. Bank, u/s (DD)	1990 Luzon, Philippines	0.02–0.06	6	15–20	0.04	7.5	Ishihara et al. [32]; Özener [19]
10	Magsaysay Bridge E. Bank, u/s (EE)		0.03–0.09	9	15–20	0.06	9.5	Ishihara et al. [32]; Özener [19]
11	Hotel Sapanca		0.05–0.11	13.4	6.2	0.08	7.0	Cetin et al. [33]; Özener [19]
12	Police station	1999 Kocaeli, Turkey	0.07–0.10	5	24.55	0.09	7.0	Cetin et al. [33]; Özener [19]
13	Soccer field		0.05–0.10	7	34	0.08	9.7	Cetin et al. [33]; Özener [19]
14	Yalova Harbor		0.14–0.20	14.53	20.8	0.17	16.3	Cetin et al. [33]; Özener [19]
15	Wufeng site C (A-A′)		0.05–0.12	3.5	25.5	0.09	6.5	Chu et al. [34]; Özener[19]
16	Wufeng site C (B-B′)	1999 Chi-Chi, Taiwan	0.10–0.23	3.5	22	0.17	5.3	Chu et al. [34]; Özener [19]
17	Wufeng site B		0.05–0.11	10	22	0.08	11.8	Chu et al. [34]; Özener [19]
18	Nantou Site N		0.16–0.23	9	16.45	0.20	10.4	Chu et al. [34]; Özener [19]
19	Norswig drive	2003 San Simeon	0.16–0.11	7.2	5	0.14	7.7	Olson and Johnson [2]; Özener [19]
20	Juanita Avenue		0.03–0.10	9.2	5	0.07	9.7	Olson and Johnson [2]; Özener [19]
21	Snow River Bridge	1964 Alaska	0.02	5–10	10–30	0.02	9.2	Özener [19]
22	Juvenile Hall	1971 San Fernando	0.02–0.08	6.9	35	0.05	9.7	Bennett [35]; Olson and Johnson [2]; Özener [19]

TABLE 2: Continued.

Case no.	Location of lateral spreading	Earthquake event	Residual shear strength ratio	SPT blow count	Fines content (%)	Median residual shear strength ratio	Equivalent clean sand blow count	Reference
23	Whiskey Sprins Fan	1983 Borah Peak	0.05–0.12	13.0	22	0.08	14.8	Andrus and Youd [36]; Olson and Johnson [2]; Ozener [19]
24	Landing Road Bridge		0.01–0.1	5–10	8–18	0.06	8.7	Olson and Johnson [2]; Ozener [19]
25	James Street Loop	1987 Edgecumbe, New Zealand	0.01–0.09	4.8–9.2	10	0.05	9.5	Olson and Johnson [2]; Özener [19]
26	Whakatane Pony Club		0.02–0.07	3–12	15–18	0.05	8.9	Olson and Johnson [2]; Özener [19]
27	Nalband Railway Station	1988 Armenia	0.08–0.10	3.6–23	25	0.05	15.3	Yegian [37]; Olson and Johnson [2]; Özener [19]
28	Farris Farm		0.04–0.13	9–24	5–38	0.09	18.3	Olson and Johnson [2]; Özener [19]
29	Leonardini Farm		0.02–0.11	10.0	10%	0.07	11.0	Olson and Johnson [2]; Özener [19]
30	Sea Mist Farm	1989 Loma Prieta	0.03–0.08	7–11	16–29	0.06	10.8	Olson and Johnson [2]; Özener [19]
31	Moss Landing, MLML Bldg., eastward spread		0.07–0.12	14.6	4	0.09	15.0	Mejia [29]; Olson and Johnson [2]; Özener [19]
32	Rudbaneh Town Canal	1990 Manjil, Iran	0.05–0.19	8.63	4.6	0.12	9.1	Ishihara et al. [38]; Yegian et al. [39]; Olson and Johnson [2]; Özener [19]
33	Magsaysay Bridge E. Bank, u/s (AA)		0.02–0.08	2–9	20–30	0.05	7.5	Ishihara et al. [40]; Olson and Johnson [2]; Özener [19]
34	Nable Street West (B-B)	1990 Luzon, Philippines	0.03–0.09	4.2–10	17.5–27.5	0.06	8.9	Ishihara et al. [40]; Olson and Johnson [2]; Özener [19]
35	Nable Street West (C-C)		0.02–0.07	4–8.5	10–20	0.05	7.6	Ishihara et al. [40]; Olson and Johnson [2]; Özener [19]
36	Pogo Chico W. Bank		0.03–0.10	6–11	11.3–20	0.07	9.9	Ishihara et al. [40]; Olson and Johnson [2]; Özener [19]
37	Balboa Blvd.		0.11–0.19	17.0	52	0.15	21.0	Holzer et al. [41]
38	Wynne Avenue	1994 Northridge	0.09–0.20	11.6	33	0.15	14.2	Holzer et al. [41]; Olson and Johnson [2]; Idriss and Boulanger [27]; Özener [19]
39	Potrero Canyon		0.09–0.21	5–17	39–75	0.15	15.28	Lumbantoruan [42]; Özener [19]
40	Seymen Tea Garden	1999 Kocaeli, Turkey	0.04–0.08	6–12	20–40	0.06	11.4	Lumbantoruan [42]; Özener [19]
41	Esme Nose		0.06–0.11	5–13	10–30	0.09	10.33	Lumbantoruan [42]; Özener [19]
42	Wufeng site C1	1999 Chi-Chi, Taiwan	0.10–0.18	11–18	22	0.14	16.3	Olson and Johnson [2]; Chu et al. [34]; Özener [19]
43	Wufeng site M		0.07–0.15	11.5	12	0.11	12.6	Olson and Johnson [2]; Chu et al. [34]; Özener [19]

$$\text{MSE} = \frac{1}{M} \sum_{i=1}^{M} (y_i - \hat{y}_i)^2, \qquad (5)$$

where MSE is the mean squared error between the target value and output value generated by the neural network, M is

the number of observations, y_i is the target value, and $\hat{y}_i$ is the output value generated by the neural network.

It can be seen from Figure 2 that the data of the predicted residual shear strength ratio lie within the range of 85% prediction. Figure 3 shows the cumulative probability of the
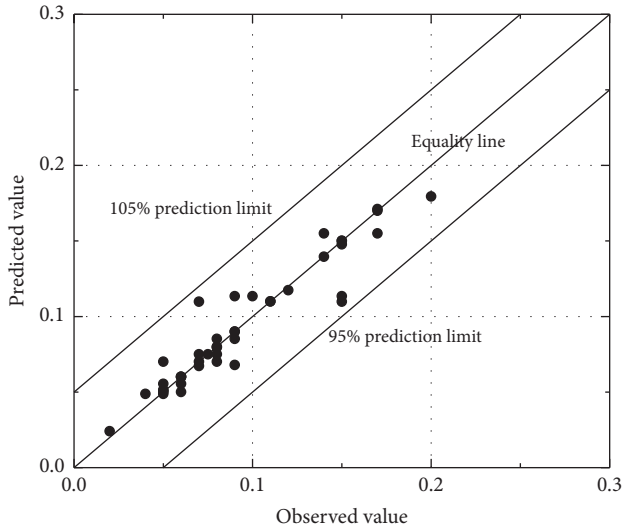
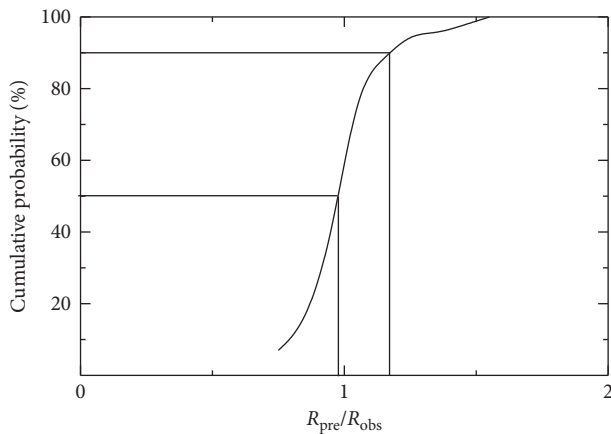FIGURE 2: The observed and predicted values of residual shear strength ratio.

FIGURE 3: Cumulative probability of the ratio (R_{pre}/R_{obs}) for Artificial Neural Network model.

ratio of the predicted residual shear strength to the observed residual shear strength for the Artificial Neural Network model, where R_{obs} is the observed residual shear strength ratio and R_{pre} is the predicted the residual shear strength ratio. The 50% cumulative probability is close to 1.0, and the 90% cumulative probability is 1.15, which is less than 1.5; thus, the model shows slight underestimation at 50% probability, and the value of the residual shear strength ratio at 90% probability indicates a small variance for the total predictions.

2.2. Prediction Comparison. Olson and Johnson [2] proposed an estimation method that was used to describe the relationship between the residual shear strength ratio (the ratio of residual shear strength to overburden stress) and the SPT blow count of the liquefied sand based on lateral spreading case histories. As shown in equation (6), S_u is the mobilized strength (i.e., residual shear strength), σ'_v is the

prefailure vertical effective stress (i.e., effective overburden stress), and $(N_1)_{60}$ is the SPT blow count of the liquefied sand. In the empirical model proposed by Olson and Johnson [2], the fines content was not considered for calculating residual shear strength ratio, and value of $(N_1)_{60}$ was assumed to be less than 16.0:

$$\frac{S_u}{\sigma'_v} = 0.03 + 0.0075\left[(N_1)_{60}\right] \pm 0.03. \tag{6}$$

The predicted residual shear strength using the Artificial Neural Network model was compared with the median residual shear strength ratio by the Olson and Johnson estimation method [2]. Figure 4 shows the predicted residual shear strength ratios by the two methods. To show the variance of the predicted residual shear strength ratio, the root mean square error (RMSE), expressed in equation (7), is used:

$$\text{RMSE} = \frac{1}{M} \sum_{i=1}^{M} y_i - x_i^2, \tag{7}$$

where RMSE is the root mean square error between the target value and output value of prediction, M is the number of observations, y_i is the target value, and x_i is the output value of prediction.

The root mean square errors (RMSE) are 0.0783 and 0.0697 for Olson and Johnson estimation method [2] and the Artificial Neural Network model, respectively. The smaller root mean square error (RMSE) from the Artificial Neural Network model indicates a better prediction is made by the Artificial Neural Network model.

2.3. Newmark Displacement of Lateral Spreading Using Residual Shear Strength. The Newmark sliding block method, which was proposed by Newmark [43], has been widely used in evaluating the permanent displacement of slopes, embankments, and landfills. The seismic lateral spreading can be used by the Newmark sliding block method [6–8], in which the soil above the liquefied soil is regarded as a perfect rigid sliding block and the yield acceleration is calculated based on the limit equilibrium analysis. The block is sliding at the constant yield acceleration. The displacement of lateral spreading begins to accumulate when the acceleration of the ground soil above the liquefied soil is greater than the yield acceleration. Once the relative velocity of the ground soil to the sliding surface, which is corresponding with the liquefied soil layer, equals to zero, the displacement of lateral spreading stops to accumulate.

The residual shear strength is the minimum shear strength that is corresponding to the worst-case scenario, so the use of residual shear strength can evaluate the post-liquefaction stability of soil deposits for lateral spreading and determine the magnitude of lateral spreading induced by liquefaction. In this section, the residual shear strength is predicted by the Artificial Neural Network model and used in the yield acceleration calculation, which is calculated by the limit equilibrium method. The lateral spreading is calculated by the Newmark sliding block method using the

● Olson and Johnson method
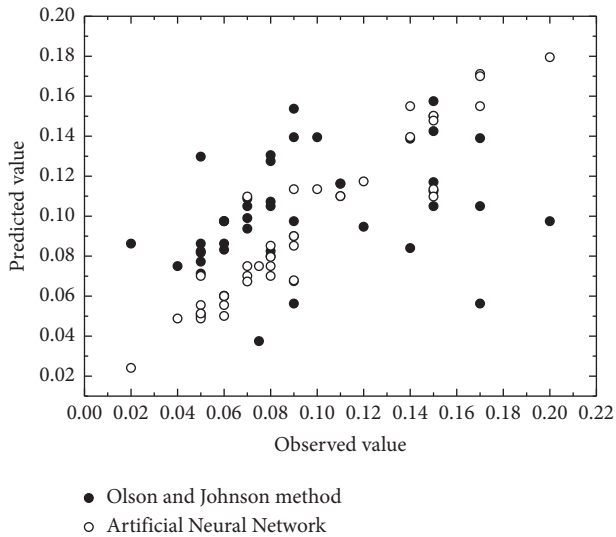○ Artificial Neural Network

FIGURE 4: The predicted values by Artificial Neural Network model and Olson and Johnson method [2].

postliquefaction yield acceleration corresponding to the residual shear strength of liquefied soil.

The high-quality case histories of lateral spreading are used to apply the Newmark sliding block method. Among the 43 case histories that are used for training the Artificial Neural Network model, the case histories with detailed site investigation and available earthquake motions are classified into the group of high-quality case histories. A total number of 23 case histories grouped as high-quality case histories are selected from Table 2 when the SPT blow count of liquefied soil, the thickness of liquefied soil, the soil profiles, soil classifications, and site-specific earthquake motions or the earthquake motions close to the site are recorded or are available in the research literature. Applying the Morgenstern-Price method, two-dimensional soil profile and the residual shear strength of liquefied soil are used in the limit equilibrium method to obtain the postliquefaction yield acceleration. The soil parameters for the nonliquefiable soil used in the limit equilibrium analysis are based on the soil classifications and reasonable values for different types of soil are assigned. In Table 3, the case histories used in the Newmark sliding block method, the PGA of the site, the effective stress, the residual shear strength of the liquefied soil predicted by the Artificial Neural Network model, and the corresponding yield acceleration from limit equilibrium method are listed for each case history. The average predicted lateral spreading for each case using the motions is also listed in Table 3. Table 4 lists the calculated lateral spreading based on the Newmark sliding block analysis and for each case history using different earthquake motions. In Table 4, the normal displacement and inverse displacement are representing the displacements for each case when the two different directions of the motion were used.

To illustrate the calculation process of applying the Newmark sliding block method in calculating lateral spreading, the case history of Treasure Island was analyzed and shown in the paper. The Treasure Island is a manmade

island created by hydraulically placing sand fill over soft sedimentary deposits. Sand boils and ground cracks induced by liquefaction were observed at the site. In Figure S1, the cross section of the subsurface used in the limit equilibrium analysis is shown. The recorded lateral spreading was 0.25 m based on Power et al. [31]. The laboratory tests were conducted on the samples from the drilled boreholes and Cone Penetration Resistance Test was conducted. The site consisted of four layers of soil from the ground surface to the bottom layer: hydraulically placed sand fill, native Yerba Buena shoals sand and clay, bay mud, and older bay sedimentary deposits. The sand fill was poorly graded silty sand with clayey sand zones. The sand fill was underlain by Yerba Buena shoals which were clayey sand with clay layers and had similar engineering characteristics to sand fill. Both the sand fill and shoal sand were loose and susceptible to the liquefaction under the earthquake. During the earthquake, the sand fill was found to liquefy. Below the Yerba Buena shoals sand, the bay mud consisted of soft to stiff olive-gray silty clay and the older bay sedimentary deposits consisted of brownish and greenish-gray, very stiff sandy, silty, and peaty clay and dense sand, which were insusceptible to lateral movements or settlements. The bay mud was normally consolidated and the strength was increased by the overburden from sand fill and shoals. The groundwater table was affected by tidal fluctuations. $(N_1)_{60}$ of the liquefiable soil on the island was between 5 and 15 blows/ft, so a median of 10.0 and a fines content of 10% were used to estimate the residual shear strength. Based on the effective stress of 43.99 kPa and the equivalent clean sand corrected SPT blow count of 11.0, the residual shear strength for the liquefied soil is 4.83 kPa. The soil parameters for nonliquefiable soil and the liquefied soil used to calculate the postliquefaction yield acceleration are listed in Table S1. A postliquefaction acceleration of 0.052 g was obtained for the Treasure Island case based on the limit equilibrium analysis, as shown in Figure S2.

The peak acceleration was estimated to be 0.16 g; two motions recorded at Yerba Buena Island from PEER (Pacific Earthquake Engineering Research Center) database [45] were used to calculate the lateral spreading induced by seismic liquefaction. The PGAs of the two motions were scaled to 0.16 g to conduct the Newmark sliding block analyses. The site classification of the soil where the strong-motion station was located was Class E, of which the shear wave velocity was less than 180 m/s based on NEHRP site classification [46]. The average displacement for the Treasure Island case is listed in Table 3 and calculated lateral spreading is listed in Table 4. The normal and inverse displacement represents the two displacements corresponding to two directions of the input motion. In Figure S3, the Newmark displacements using the two motions are plotted. Figure S3(a) showed the variation of normal-direction displacement versus time and Figure S3(b) showed the variation of inverse-direction displacement versus time for the Treasure Island case. The average lateral spreading was 0.13 m, which was 52% of the recorded lateral spreading. Table S2 lists the earthquake motions used in the Newmark sliding block analysis. In Table S2, there are 41 total motions used for the analyses, which were available in the motion

TABLE 3: The average predicted and recorded lateral spreading for each case history.

Case	Site location	Earthquake event	PGA (g)	Effective stress (kPa)	Residual shear strength (kPa)	Yield acceleration (g)	Predicted lateral spreading (m)	Recorded lateral spreading (m)	Displacement ratio	Reference
1	Juvenile hall	San Fernando (1971)	0.7	86.56	5.80	0.06	1.72	1.5	1.15	Bennett [35]
2	Heber road	Imperial Valley (1979)	0.8	61.97	4.65	0.03	1.68	2.1	0.80	Castro [23]; Youd and Bennett [24]
3	Whiskey Springs fan	Borah Peak (1983)	0.6	107.41	8.38	0.023	1.50	0.75	2.00	Andrus and Youd [36]
4	Wildlife site	Superstition Hills (1987)	0.21	61.65	4.62	0.01	1.70	0.18	9.44	Holzer et al. [25]; Boulanger et al. [26]; Idriss and Boulanger [27]
5	Moss Landing Bldg. 4	Loma Prieta (1989)	0.25	63.73	10.90	0.11	0.05	0.28	0.18	Boulanger et al. [28]
6	Moss Landing Bldg. 3	Loma Prieta (1989)	0.25	84.03	16.05	0.10	0.07	0.25	0.28	Boulanger et al. [28]
7	MLML eastward (A-A)	Loma Prieta (1989)	0.25	71.87	8.19	0.08	0.11	0.45	0.24	Mejia [29]
8	MLML eastward (B-B)	Loma Prieta (1989)	0.25	139.86	15.94	0.14	0.02	0.45	0.04	Mejia [29]
9	Leonardini Farm	Loma Prieta (1989)	0.16	29.97	4.92	0.23	N.A.	0.25	N.A.	Charlie et al. [44]
10	Treasure Island	Loma Prieta (1989)	0.16	43.99	4.83	0.052	0.13	0.25	0.52	Power et al. [31]
11	Rudbaneh Town Canal	Manjil, Iran (1990)	0.15	232.52	27.30	0.05	0.70	1.0	0.70	Yegian et al. [39]
12	Balboa Blvd.	Northridge (1994)	0.85	109.12	16.37	0.16	0.64	0.5	1.28	Holzer et al. [41]
13	Wynne Ave	Northridge (1994)	0.51	124.28	18.64	0.168	0.16	0.15	1.07	Holzer et al. [41]; Olson and Johnson [2]; Idriss and Boulanger [27]
14	Wufeng site C (A-A)	Chi-Chi, Taiwan (1999)	0.81	80.18	5.45	0.09	2.43	2.05	1.19	Chu et al. [34]
15	Wufeng site C (B-B)	Chi-Chi, Taiwan (1999)	0.81	80.81	13.74	0.30	0.09	0.49	0.18	Chu et al. [34]
16	Wufeng site C1	Chi-Chi, Taiwan (1999)	0.81	76.79	11.90	0.12	1.41	1.24	1.14	Chu et al. [34]
17	Wufeng site B	Chi-Chi, Taiwan (1999)	0.81	67.64	5.41	0.06	4.38	2.96	1.48	Chu et al. [34]
18	Wufeng site M	Chi-Chi, Taiwan (1999)	0.81	59.83	6.58	0.15	0.84	1.62	0.52	Chu et al. [34]

TABLE 3: Continued.

Case	Site location	Earthquake event	PGA (g)	Effective stress (kPa)	Residual shear strength (kPa)	Yield acceleration (g)	Predicted lateral spreading (m)	Recorded lateral spreading (m)	Displacement ratio	Reference
19	Nantou Site N	Chi-Chi, Taiwan (1999)	0.42	30.59	5.49	0.11	0.27	0.25	1.08	Chu et al. [34]
20	Hotel Sapanca	Kocaeli, Turkey (1999)	0.4	35.77	5.34	0.055	1.40	2.0	0.70	Cetin et al. [33]
21	Police station	Kocaeli, Turkey (1999)	0.4	29.73	2.53	0.02	4.44	2.4	1.85	Cetin et al. [33]
22	Soccer field	Kocaeli, Turkey (1999)	0.4	43.48	3.05	0.125	0.20	1.2	0.17	Cetin et al. [33]
23	Yalova Harbor	Kocaeli, Turkey (1999)	0.3	82.03	12.71	0.11	0.30	0.3	1.00	Cetin et al. [33]

TABLE 4: Summarization of calculated lateral spreading.

Case no.	Motion name	Normal displacement (m)	Inverse displacement (m)	Observed (m)
1	PAS-000	1.48	1.41	1.50
	PAS-090	2.12	1.98	
	PDL-120	2.63	2.47	
	PDL-210	0.86	0.85	
2	AGR-003	1.97	1.61	2.10
	BCR-140	1.79	1.91	
	BCR-230	1.82	1.62	
	SHP-270	1.36	1.39	
3	BOR000	1.70	1.88	0.75
	BOR090	1.29	1.12	
4	WSM-090	1.67	1.99	0.18
	WSM-180	1.41	1.73	
5	GOF-160	0.03	0.02	0.28
	GOF-250	0.01	0.01	
	HCH-090	0.06	0.01	
	HCH-180	0.23	0.11	
	HDA-165	0.02	0.04	
	HDA-225	0.06	0.04	
6	GOF-160	0.04	0.03	0.25
	GOF-250	0.02	0.02	
	HCH-090	0.07	0.01	
	HCH-180	0.29	0.14	
	HDA-165	0.03	0.05	
	HDA-225	0.07	0.05	
7	AND-250	0.07	0.03	0.45
	AND-340	0.07	0.04	
	G02-000	0.03	0.06	
	G02-090	0.08	0.09	
	HCH-090	0.12	0.06	
	HCH-180	0.45	0.24	
8	AND-250	0.01	0.01	0.45
	AND-340	0.01	0.01	
	G02-000	0.00	0.01	
	G02-090	0.01	0.01	
	HCH-090	0.04	0.00	
	HCH-180	0.08	0.05	

Table 4: Continued.

Case no.	Motion name	Normal displacement (m)	Inverse displacement (m)	Observed (m)
	G02-000	0.00	0.00	
9	G02-090	0.00	0.00	0.25
	HCH-090	0.00	0.00	
	HCH-180	0.00	0.00	
10	TRI-000	0.06	0.14	0.25
	TRI-090	0.11	0.20	
11	MANJIL-188040	0.41	0.39	1.00
	MANJIL-188310	1.11	0.86	
	PAR-L	0.55	0.70	
12	PAR-T	1.13	1.05	0.50
	SYL-090	0.64	0.22	
	SYL-360	0.52	0.31	
	CNP-106	0.13	0.09	
	CNP-196	0.12	0.19	
13	SCE-288	0.11	0.17	0.15
	STC-090	0.08	0.09	
	STC-180	0.29	0.28	
14	TCU065-000	3.08	2.95	2.05
	TCU065-090	2.12	1.59	
15	TCU065-000	0.14	0.16	0.49
	TCU065-090	0.08	0.00	
16	TCU065-000	1.96	1.75	1.20
	TCU065-090	1.14	0.79	
17	TCU065-000	5.14	4.89	2.96
	TCU065-090	3.89	3.59	
18	TCU065-000	1.27	1.05	1.62
	TCU065-090	0.65	0.39	
19	TCU076-000	0.27	0.14	0.25
	TCU076-090	0.34	0.33	
20	YPT-060	2.19	1.60	2.00
	YPT-330	0.87	0.94	
21	YPT-060	6.26	5.98	2.40
	YPT-330	3.02	2.49	
22	YPT-060	0.30	0.19	1.20
	YPT-330	0.13	0.16	
23	YPT-060	0.48	0.28	0.30
	YPT-330	0.21	0.23	

database of PEER (Pacific Earthquake Engineering Research Center). For each motion, the distance from the motion to the site and the site classification are included in Table S2.

For Case 9, the postliquefaction yield acceleration is greater than the reported PGA [44]; thus the analysis was not conducted for this case. The predicted lateral spreading for 11 case histories is greater than that for the observed values, and the ratios of the rest of case histories are larger or equal to 1.0. The average ratio of the 22 case histories is 1.23, which indicates that the Newmark sliding displacement of lateral spreading using the residual shear strength overall is greater than the observed lateral spreading. For the case of Wildlife Site, the displacement ratio is 9.44, of which the value may be too large compared to the other case histories. If the Wildlife Site case is not accounted for in the calculation of the average ratio of lateral spreading, the average value for the rest of the 21 case histories is 0.836, with a standard deviation of 0.56. Due to the limited case histories, the standard deviation of the lateral spreading ratio is still large. In Figure 5, the predicted lateral spreading and observed lateral spreading are plotted, and the prediction limit lines representing the

lateral spreading ratio of 0.5 and 2 are shown, respectively. There are 6 out of the 22 case histories having a lateral spreading ratio less than 0.5, and 15 out the 22 case histories having a lateral spreading ratio less than 2.0 and greater than 0.5. The cumulative distribution of the lateral spreading ratio calculated with the Newmark sliding block method can be obtained if the frequency counts of the lateral spreading ratios are obtained and proper distribution relationship is used. As the lateral spreading ratio is always greater than 0.0, the truncated normal distribution is derived from a normally distributed random variable by defining the range of the random variable. Assume that the range of the lateral spreading is from 0.0 to infinite; when applying a truncated normal distribution to describe the distribution of lateral spreading ratio, a probability of 98% is obtained if a lateral spreading ratio of 2.0 is expected.

2.4. Fitting Relationship between the Residual Shear Strength Ratio and SPT Value. A fitting curve is proposed to describe the relationship between the residual shear strength ratio
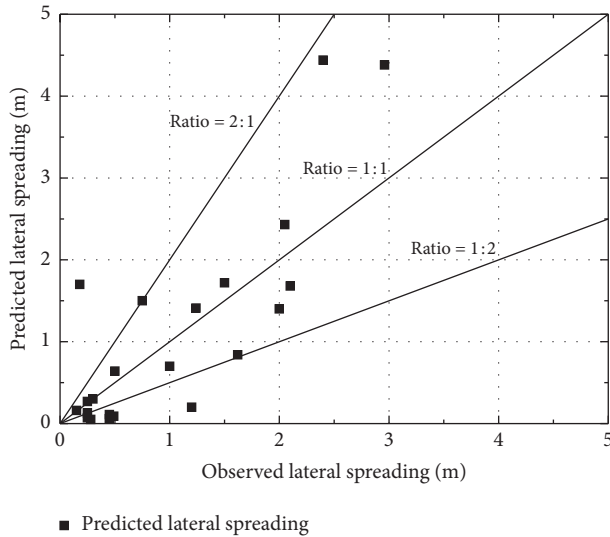
FIGURE 5: Predicted lateral spreading versus observed lateral spreading.

FIGURE 6: The relationship between the residual shear strength ratio and the equivalent clean sand corrected SPT blow count.

and the equivalent clean sand corrected SPT blow count in Figure 6. The average value of the residual shear strength ratios is used for the condition that one equivalent clean sand corrected SPT blow count of the liquefied soil is corresponding to various ratios. An exponential function is used to correlate the residual shear strength ratio to equivalent clean sand corrected SPT blow count of liquefied soil. In equation (8), the exponential function used for fitting is shown:

$$R = \exp\left[0.0004 * (N_1)_{60-cs} + 0.0008 * ((N_1)_{60-cs})^2 - 2.170\right], \tag{8}$$

where R is the residual shear strength ratio, $(N_1)_{60-cs}$ is the equivalent clean sand corrected SPT blow count of the liquefied soil, and the fines content correction values of the SPT blow count are based on values recommend by Seed [1].

3. Discussion

Although the residual shear strength ratio was successfully predicted by the Artificial Neural Network model, with a good correlation coefficient for the predicted values, there are still uncertainties that are limiting the further application of Artificial Neural Network. The number of case histories used in the analysis is limited. There are 43 case histories in total in the development of Artificial Neural Network model, and the median values for the residual shear strength ratio and the equivalent clean sand corrected SPT blow count cannot represent the variance of the soil parameters and may induce errors when training Artificial Neural Network model. Furthermore, the value of equivalent clean sand corrected SPT blow count varies when different instruments and testing standards are used.

The nonlinear dynamic response of the sliding mass above the sliding surface has been analyzed by several researchers [47–49], and the effects of the deformable sliding mass and bedrock or the soil below the sliding surface need
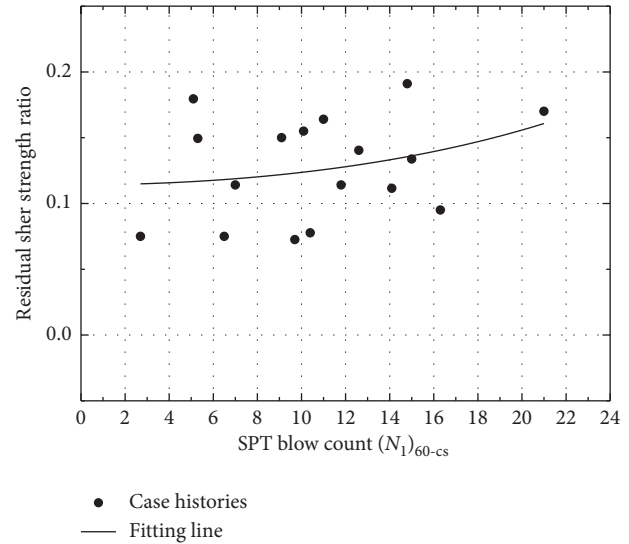
to be considered if the displacement of lateral spreading or other Earth structures are desired. Regarding the conventional Newmark sliding block method used in this paper, the intact soil above the liquefied soil is assumed to be a perfectly rigid block, the sliding surface used in the limit equilibrium analysis may not be consistent with the location where the liquefaction occurred, and the deformation of the sliding mass is omitted, so the inaccuracy of the predicted lateral spreading is too attributed to the fact that the dynamic response of the sliding mass is not considered. Moreover, the yield acceleration changes as the inclination (downward movement of sliding mass) decreases [50, 51], the change of the geometry for the sliding mass [52] is affecting the yield acceleration, and these two factors can be referred to as the ration effects of the sliding mass. Still, the perfectly-rigid-block assumption of the conventional Newmark sliding block method will yield to a constant yield acceleration, and the omitting of rotation effects of the sliding mass would result in the uncertainty of the lateral spreading and cause the inaccurate predictions. The measurement of the observed lateral spreading reported in the research literature and the ground motions used in the analyses are also contributing to the uncertainties of the predicted lateral spreading.

The Artificial Neural Network model proposed in this paper is providing an insight of the residual shear strength of lateral spreading case histories, while various uncertain factors used in terms of developing the Artificial Neural model need to be paid more attention and the residual shear strength of liquefied soil needs to be used with caution when calculating lateral spreading with Newmark sliding block method.

4. Conclusions

The evaluation of lateral spreading induced by liquefaction requires the evaluation of the residual strength of the liquefied soil, but it is difficult to determine the residual shear

strength of liquefiable soil in engineering practice. In this paper, the residual shear strength ratio of liquefied soil is predicted by the Artificial Neural Network model based on the case histories of lateral spreading. To apply the prediction model of the Artificial Neural Network model, a set of high-quality case histories was reanalyzed. The lateral spreading was calculated with Newmark sliding block method and the yield acceleration for each case was calculated with limit equilibrium method using the residual shear strength of liquefied soil predicted by the Artificial Neural Network model. Based on the predictions of residual shear strength ratio predicted by the Artificial Neural Network model and lateral spreading calculated with the Newmark sliding block method, the following conclusions can be drawn.

By taking account of the residual shear strength ratio and the equivalent clean sand corrected SPT blow count of liquefied soil for 43 case histories of lateral spreading, an Artificial Neural Network model was proposed in terms of residual shear strength ratio, the correlation coefficient of the proposed Artificial Neural Network model is 0.92, and the mean squared error (MSE) is 0.001. The value of the residual shear strength ratio at 90% probability indicates a small variance for the total predictions when applying the proposed Artificial Neural Network model.

Comparing the Artificial Neural Network model to the residual shear strength model by Olson and Johnson, a root mean square error (RMSE) of 0.0697 shows that the proposed Artificial Neural Network model predicts a better ratio compared to Olson and Johnson model, the root mean square error (RMSE) of which is 0.0783.

The lateral spreading calculated by Newmark sliding block method and the postliquefaction yield acceleration based on the residual shear strength from the proposed Artificial Neural Network model for high-quality case histories show that the average ratio of lateral spreading (the ratio of predicted lateral spreading to observed lateral spreading) is 0.836, with a standard deviation of 0.56. When a truncated normal distribution is used to describe the distribution of lateral spreading ratio, the confidence level with a probability of 98% is obtained if a lateral spreading ratio of 2.0 is expected. Based on the high-quality case histories, an exponential fitting curve is proposed to describe the relationship between the residual shear strength ratio and the equivalent clean sand corrected SPT blow count.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

Supplementary Materials

Figure S1: cross section of the subsurface used in the limit equilibrium analysis. Figure S2: limit equilibrium analysis of post-liquefaction yield acceleration. Figure S3: Newmark displacement using the two motions: (a) variation of normal-direction displacement versus time; (b) variation of inverse-direction displacement versus time. Table S1: soil parameters used in the limit equilibrium analysis of post-liquefaction yield acceleration. Table S2: earthquake motions used for each case history. (*Supplementary Materials*)

References

[1] H. B. Seed, "Design problems in soil liquefaction," *Journal of Geotechnical Engineering*, vol. 113, no. 8, pp. 827–845, 1987.

[2] S. M. Olson and C. I. Johnson, "Analyzing liquefaction-induced lateral spreads using strength ratios," *Journal of Geotechnical and Geoenvironmental Engineering*, vol. 134, no. 8, pp. 1035–1049, 2008.

[3] M. H. Hassoun, *Fundamentals of Artificial Neural Networks*, MIT Press, Cambridge, MA, USA, 1995.

[4] W. S. Noble, "What is a support vector machine?" *Nature Biotechnology*, vol. 24, no. 12, pp. 1565–1567, 2006.

[5] J. R. Koza, *Genetic Programming*, MIT Press Cambridge, Cambridge, MA, USA, 1994.

[6] M. H. Baziar, R. Dobry, and A. W. Elgamal, "Engineering evaluation of permanent ground deformations due to seismically induced liquefaction," Technical Report NCEER-92-0007, p. 269, National Center for Earthquake Engineering Research, Buffalo, NY, USA, 1992.

[7] V. M. Taboada, T. Abdoun, and R. Dobry, "Prediction of liquefaction-induced lateral spreading by dilatant sliding block model calibrated by centrifuge tests," in *Proceedings of the 11th World Conf. On Earthquake Engineering*, Pergamon Oxford, UK, 1996.

[8] National Academies of Sciences, *Engineering, and M. State of the Art and Practice in the Assessment of Earthquake-Induced Soil Liquefaction and its Consequences*, The National Academies Press, Washington, DC, USA, 2016.

[9] Rocscience, *Slide v5.0-2D limit equilibrium slope Stability Analysis*, Rocscience Inc, Toronto, Canada, 2005.

[10] R. W. Jibson, E. M. Rathje, M. W. Jibson, and Y. W. Lee, *SLAMMER: Seismic Landslide Movement Modeled Using Earthquake Records*, US Geological Survey, Reston, VA, USA, 2013.

[11] H. B. Seed, R. B. Seed, L. F. Harder, and H. L. Jong, "Re-evaluation of the slide in the Lower San Fernando Dam in the 1971 San Fernando earthquake," WES Contract Report GL-89-2, United States Army Corps of Engineers, Vicksburg, MS, USA, 1998.

[12] I. M. Idriss, "Evaluation of liquefaction potential, consequences and mitigation an update," in *Proceedings of the Presentation Notes for Geotechnical Society Meeting*, Vancouver, Canada, February 1998.

[13] T. D. Stark and G. Mesri, "Undrained shear strength of liquefied sands for stability analysis," *Journal of Geotechnical Engineering*, vol. 118, no. 11, pp. 1727–1747, 1992.

[14] S. M. Olson and T. D. Stark, "Liquefied strength ratio from liquefaction flow failure case histories," *Canadian Geotechnical Journal*, vol. 39, no. 3, pp. 629–647, 2002.

[15] I. M. Idriss and R. W. Boulanger, "SPT-and CPT-based relationships for the residual shear strength of liquefied soils," in *Earthquake geotechnical engineering*, K. D. Pitilakis, Ed., vol. 6, pp. 1–22, Springer, Dordrecht, Netherland, 2007.

[16] I. M. Idriss and R. W. Boulanger, "Lecture: 2nd Ishihara lecture: SPT- and CPT-based relationships for the residual shear strength of liquefied soils," *Soil Dynamics and Earthquake Engineering*, vol. 68, pp. 57–68, 2015.

[17] S. L. Kramer, "Evaluation of liquefaction hazards in Washington state," Washington State Department of Transportation, Office of Research and Library Services, Washington, DC, USA, WA-RD 668.1, 2008.

[18] S. L. Kramer and C.-H. Wang, "Empirical model for estimation of the residual strength of liquefied soil," *Journal of Geotechnical and Geoenvironmental Engineering*, vol. 141, no. 9, p. 04015038, 2015.

[19] P. Özener, "Estimation of residual shear strength ratios of liquefied soil deposits from shear wave velocity," *Earthquake Engineering and Engineering Vibration*, vol. 11, no. 4, pp. 461–484, 2012.

[20] M. H. Beale, M. T. Hagan, and H. B. Demuth, *Neural Network Toolbox*, pp. 77–81, MathWorks Inc, Natick, MA, USA, 2010.

[21] E. Sariev and G. Germano, "Bayesian regularized artificial neural networks for the estimation of the probability of default," *Quantitative Finance*, vol. 20, no. 2, pp. 311–328, 2020.

[22] A. P. Davis Jr., S. J. Poulos, and G. Castro, "Strengths backfigured from liquefaction case histories," in *Proceedings of the Second International Conference on Case Histories in Geotechnical Engineering*, Missouri University of Science and Technology (formerly the University of Missouri-Rolla), St. Louis, MI, USA, June 1988.

[23] G. Castro, "Empirical methods in liquefaction evaluation," in *Proceedings of the Proceedings of the 1st Annual Leonardo Zeevaert International Conference*, vol. 1, pp. 1–41, Mexico City, Mexico, 1995.

[24] T. L. Youd and M. J. Bennett, "Liquefaction sites, imperial valley, California," *Journal of Geotechnical Engineering*, vol. 109, no. 3, pp. 440–457, 1983.

[25] T. L. Holzer, T. C. Hanks, and T. L. Youd, "Dynamics of liquefaction during the 1987 superstition hills, California, earthquake," *Science*, vol. 244, no. 4900, pp. 56–59, 1989.

[26] R. W. Boulanger, D. W. Wilson, and I. M. Idriss, "Examination and reevalaution of spt-based liquefaction triggering case histories," *Journal of Geotechnical and Geoenvironmental Engineering*, vol. 138, no. 8, pp. 898–909, 2012.

[27] I. M. Idriss and R. W. Boulanger, "SPT-based liquefaction triggering procedures," Report No. UCD/CGM-10/02, University of California at Davis, Davis, CA, USA, 2010.

[28] R. W. Boulanger, L. H. Mejia, and I. M. Idriss, "Liquefaction at moss landing during loma prieta earthquake," *Journal of Geotechnical and Geoenvironmental Engineering*, vol. 123, no. 5, pp. 453–467, 1997.

[29] L. H. Mejia, "Liquefaction at moss landing," *Technical Report*, pp. 129–150, U.S. Geological Survey Open-File Report 00-444, United States Government Printing Office, Washington, DC, USA, 1998.

[30] T. L. Holzer, J. C. Tinsley, M. J. Bennett, and C. S. Mueller, "Observed and predicted ground deformation-Miller Farm lateral spread, Watsonville, California," vol. 94, Technical Report NCEER, pp. 79–99, US National Center for Earthquake Engineering Research (NCEER), Buffalo, NY, USA, 1994.

[31] M. S. Power, J. A. Egan, M. L. TraubenikF, and J. R. Faris, "Liquefaction at naval station treasure island and design of mitigating measures," *Technical Report*, pp. 87–121, U. S. Geological Survey Open-File Report 00-444, United States Government Printing Office, Washington, DC, USA, 1998.

[32] K. Ishihara, A. A. Acacio, and I. Towhata, "Liquefaction-induced ground damage in dagupan in the july 16, 1990 Luzon earthquake," *Soils and Foundations*, vol. 33, no. 1, pp. 133–154, 1993.

[33] K. O. Cetin, T. L. Youd, R. B. Seed et al., "Liquefaction-induced ground deformations at hotel sapanca during kocaeli (izmit),Turkey earthquake," *Soil Dynamics and Earthquake Engineering*, vol. 22, no. 9-12, pp. 1083–1092, 2002.

[34] D. B. Chu, J. P. Stewart, T. L. Youd, and B. L. Chu, "Liquefaction-induced lateral spreading in near-fault regions during the 1999 Chi-Chi, Taiwan earthquake," *Journal of Geotechnical and Geoenvironmental Engineering*, vol. 132, no. 12, pp. 1549–1565, 2006.

[35] M. J. Bennett, "Liquefaction analysis of the 1971 ground failure at the san fernando valley juvenile Hall, California," *Environmental & Engineering Geoscience*, vol. xxvi, no. 2, pp. 209–226, 1989.

[36] R. D. Andrus and T. L. Youd, "Subsurface investigation of a liquefaction-induced lateral spread, thousand springs valley, Idaho," Master thesis, Brigham Young Univeristy, Provo, UT, USA, 1987.

[37] M. K. Yegian, V. G. Ghahraman, and R. N. Harutiunyan, "Liquefaction and embankment failure case histories, 1988 Armenia earthquake," *Journal of Geotechnical Engineering*, vol. 120, no. 3, pp. 581–596, 1994.

[38] K. Ishihara, S. M. Haeri, A. A. Moinfar, I. Towhata, and S. Tsujino, "Geotechnical aspects of the june 20, 1990 Manjil earthquake in Iran," *Soils and Foundations*, vol. 32, no. 3, pp. 61–78, 1992.

[39] M. K. Yegian, V. G. Ghahraman, M. A. A. Nogole-Sadat, and H. Daraie, "Liquefaction during the 1990 Manjil, Iran, earthquake, II: case history analyses," *Bulletin of the Seismological Society of America*, vol. 85, pp. 83–92, 1995.

[40] K. Ishihara, "Characterization of cyclic behavior of sand and post-seismic stability analyses," in *Proceedings of the 9th Asian Regional Conference on Soil Mechanics and Foundation Engineering*, Springer, Bangkok, Thailand, December 1991.

[41] T. L. Holzer, M. J. Bennett, D. J. Ponti, and J. C. Tinsley III, "Liquefaction and soil failure during 1994 Northridge earthquake," *Journal of Geotechnical and Geoenvironmental Engineering*, vol. 125, no. 6, pp. 438–452, 1999.

[42] P. M. H. Lumbantoruan, "Probabilistic postliquefaction residual shear strength analyses of cohesionless soil deposits: application to the Kocaeli (1999) and Duzce (1999) earthquakes," Doctoral thesis, Virginia Polytechnic Institute, Blacksburg, VA, USA, 2005.

[43] N. M. Newmark, "Effects of earthquakes on dams and embankments," *Géotechnique*, vol. 15, no. 2, pp. 139–160, 1965.

[44] W. A. Charlie, D. O. Doehring, J. P. Brislawn, and H. Hassen, "Direct measurement of liquefaction potential in soils of Monterey County, California," *Technical Report*, pp. 181–222, U.S. Geological Survey Open-File Report 00-444, Washington, DC, USA, 1998.

[45] Y. Bozorgnia, N. A. Abrahamson, L. A. Atik et al., "NGA-West2 research project," *Earthquake Spectra*, vol. 30, no. 3, pp. 973–987, 2014.

[46] Building Seismic Safety Council, *NEHRP Recommended Seismic Provisions for New Buildings and Other structures*, FEMA P-1050-1, p. 750, Federal Emergency Management

Agency of the U.S. Department of Homeland Security, Washiongton D.C., USA, 2009.

[47] L. C. Katsenis, C. A. Stamatopoulos, V. P. Panoskaltsis, and B. Di, "Prediction of large seismic sliding movement of slopes using a 2-body non-linear dynamic model with a rotating stick-slip element," *Soil Dynamics and Earthquake Engineering*, vol. 129, p. 105953, 2020.

[48] E. M. Rathje and J. D. Bray, "An examination of simplified earthquake-induced displacement procedures for earth structures," *Canadian Geotechnical Journal*, vol. 36, no. 1, pp. 72–87, 1999.

[49] G. Tropeano, A. Chiaradonna, A. d'Onofrio, and F. Silvestri, "A numerical model for non-linear coupled analysis of the seismic response of liquefiable soils," *Computers and Geotechnics*, vol. 105, pp. 211–227, 2019.

[50] Y. Jafarian and A. Lashgari, "Simplified procedure for coupled seismic sliding movement of slopes using displacement-based critical acceleration," *International Journal of Geomechanics*, vol. 16, no. 4, Article ID 04015101, 2016.

[51] C. A. Stamatopoulos, "Sliding system predicting large permanent co-seismic movements of slopes," *Earthquake Engineering & Structural Dynamics*, vol. 25, no. 10, pp. 1075–1093, 1996.

[52] C. A. Stamatopoulos, C. Mavromihalis, and S. Sarma, "Correction for geometry changes during motion of sliding-block seismic displacement," *Journal of Geotechnical and Geoenvironmental Engineering*, vol. 137, no. 10, pp. 926–938, 2011.