IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 9 July 2024, accepted 14 July 2024, date of publication 17 July 2024, date of current version 11 September 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3430075

RESEARCH ARTICLE

Performance Comparison of Prediction of Hydraulic Jump Length Under Multiple Neural Network Models

ZIYUAN XU[1], ZIRUI LIU[2], AND YINGZI PENG[2]
[1]Civil and Hydraulic Engineering, Qinghai University, Xining 810016, China
[2]Hunan Diantou Education Technology Company Ltd., Changsha, Hunan 410000, China
Corresponding author: Yingzi Peng (pengyingzi@diantouedu.com)

ABSTRACT Hydraulic jump is a common physical phenomenon in the field of hydraulic engineering. The essence of hydraulic jump is the conversion and dissipation of a large amount of energy due to the interaction between vortex structures, mainly released in the form of turbulence and water waves. This process significantly reduces the kinetic energy of water flow, thereby mitigating downstream erosion and protecting hydraulic structures, which in turn extends their service life. As a crucial factor in the energy dissipation design of discharge structures, the length of the hydraulic jump is influenced by various factors, including flow velocity, upstream and downstream water depths, riverbed roughness height, and Froude number. In this study, we applied dimensional analysis to identify the key parameters influencing hydraulic jumps on the dataset provided by literature. We utilized a multi-task learning strategy, incorporating a shared feature extraction layer for characteristic modeling of hydraulic jumps within Physics-Informed Neural Networks (PINNs). Furthermore, we compared the performance of PINNs with other data-driven models such as Deep Neural Networks (DNNs), Convolutional Neural Networks (CNNs), and Transformers. The results demonstrated that these models are effective in estimating the length of hydraulic transitions and distinguishing between steady and unsteady hydraulic jump processes. Notably, the PINNs model exhibited better performance than other models, achieving an $\mathbf{R^2}$ score of **0.8818**, RMSE of 4.4627(cm), MAE of 3.3784(cm), **precision** of **0.9677** and recall of 0.9677 on the test set. These findings are significant for elucidating the characteristics and effects of hydraulic jumps in hydraulic structures, providing a scientific basis for the safe operation and design of practical hydraulic engineering projects.

INDEX TERMS Hydraulic jump, feature prediction, PINNs, neural networks.

I. INTRODUCTION

In fluid mechanics, the energy dissipation phenomenon of a hydraulic jump refers to the reduction of water flow's kinetic energy during a sudden transition from supercritical to subcritical flow, occurring when fluids of different velocities or depths interact [1]. The efficiency and morphology of hydraulic jumps are influenced by various factors, including flow velocity, upstream and downstream water depths, riverbed roughness, and Froude number. Accurate prediction of hydraulic jump characteristics, particularly the jump length, is essential for optimizing engineering designs and ensuring riverbed stability [2], [3], [4]. Beyond erosion prevention, hydraulic jumps also serve other purposes, such as water aeration and chemical reagent mixing, which can enhance water quality [5], [6].

Current research on hydraulic jump prediction can be categorized into three main approaches: statistical and empirical models [7], [8], [9], [10], numerical simulation models [11], [12], and machine learning models [13], [14], [15], [16], [17], [18]. Statistical and empirical models often utilize regression analysis and time series analysis to predict

The associate editor coordinating the review of this manuscript and approving it for publication was Prakasam Periasamy.

the probability or trend of future hydraulic jumps. Numerical simulation models solve fluid equations numerically to simulate hydraulic jumps under various fluid environments and predict their occurrence. Machine learning models learn patterns and rules of hydraulic jumps from data using algorithms.

In recent decades, the rapid development of machine learning and artificial intelligence techniques has greatly advanced the study of hydraulic jump phenomena. Traditional physical and mathematical models have made progress in predicting hydraulic jump characteristics but face challenges when dealing with complex riverbeds and nonlinear flows. In comparison, machine learning-based approaches have shown higher accuracy and efficiency in studying hydraulic jump characteristics. For example, support vector machines (SVM) have been successfully applied to predict the characteristics of free and submerged hydraulic jumps, demonstrating more accurate results than traditional models [15]. Additionally, nonlinear regression models and advanced machine learning techniques, such as ANFIS and LASSO [16], have proven effective in predicting the conjugate depth ratio of hydraulic jumps with significantly higher accuracy than traditional linear regression methods.

From the 1980s to the 1990s, efficient learning algorithms and novel network structures emerged rapidly, leading to significant progress in artificial intelligence [19], [20]. With the improvement of computational power, more sophisticated deep learning algorithms were developed, such as Convolutional Neural Networks (CNNs) [21] and Deep Belief Networks [22]. In recent years, researchers have applied deep learning techniques to various aspects of fluid dynamics, achieving promising results. CNNs have been used for flow feature detection, pressure distribution prediction, and real-time prediction of non-uniform steady laminar flows, demonstrating higher accuracy and transfer learning potential compared to traditional CFD methods [23], [24], [25].

Moreover, deep neural networks (DNNs) have shown significant effectiveness in improving turbulence models [26]. When combined with computational fluid dynamics (CFD), DNNs exhibit high accuracy and efficiency in predicting various fluid dynamic parameters of fluidized beds, such as solid velocity, gas volume fraction, and pressure drop under different operating conditions [27]. These studies highlight the potential of DNN models in reducing computational resources and time while demonstrating their stability in predicting fluidized bed behavior. Transformers [28] have also shown remarkable performance in flow field prediction and reconstruction tasks, exhibiting lower mean squared errors and better generalization capabilities compared to supervised learning methods [29], [30].

Furthermore, Physics-Informed Neural Networks (PINNs), which incorporate physical laws into neural networks, have been proposed to improve the efficiency and accuracy of processing complex system problems, such as solving the Reynolds-averaged Navier-Stokes (RANS) equations for incompressible turbulent flows without the need for specific turbulence models or assumptions [31], [32]. This approach allows for more robust and generalizable models by ensuring that the physical laws governing the system are inherently respected during the learning process. The application of PINNs has shown promising results across various domains, including fluid dynamics, where they provide a significant advantage in terms of computational efficiency and accuracy compared to traditional data-driven models [33], [34].

This paper employs a multi-task learning strategy for deep learning models. By designing a shared feature extraction layer to capture the general features of the input data, the model branches into two task-specific substructures: one is a regression branch for predicting continuous variables, and the other is a classification branch. The regression branch uses a linear output layer, while the classification branch employs a softmax activation function to predict multiclass labels.

To explore the adaptability of machine learning in hydraulic jump prediction, Physics-Informed Neural Networks (PINNs) are utilized to implement physics information-driven modeling. Simultaneously, the performance of PINNs is compared with other data-driven models, such as Deep Neural Networks (DNNs), Convolutional Neural Networks (CNNs), and Transformers models. The study further investigates the prediction accuracy of multiple models for hydraulic jump length and the steady and unsteady stages of the hydraulic jump, aiming to provide a solid scientific basis for future hydraulic engineering design, construction, and management.

In previous research, there has been little comprehensive study on estimating jump length and the development stages of hydraulic jumps using multiple models such as DNNs, CNNs, Transformers, and PINNs, especially employing PINNs for hydraulic jump modeling. Therefore, this is one of the most significant academic contributions of this research. The relevant research results can provide important theoretical support for scientific decision-making in hydraulic engineering design, construction, and management, and have significant practical engineering implications.

II. FOUNDATION OF METHODS AND DATA

A. HYDRAULIC JUMP THEORY

Hager et al. citeb35 classified the classic hydraulic jump into two types: developed and no-developed hydraulic jumps. The developed hydraulic jump is relatively smooth and stable. Slightly downstream of the toe, where the water depth is h1, the forward flow approaches the bottom and further disperses downstream. A stagnation point can be clearly identified at the end of the hydraulic jump. Due to typical boiling phenomena, air bubbles continuously rise at the end of the roller, with only minimal surface waves entering the tailwater.

Compared to the developed hydraulic jump, the no-developed hydraulic jump possesses more kinetic energy, and large-scale vortices can be visibly observed at the

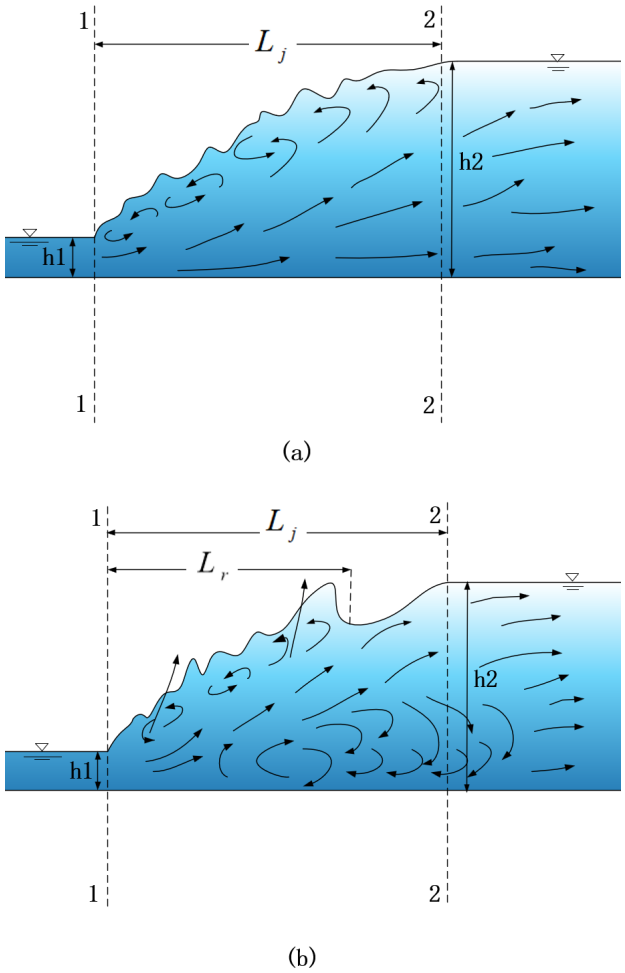


FIGURE 1. Smooth bed (a) Developed hydraulic jump (b) Non-developed hydraulic jump.

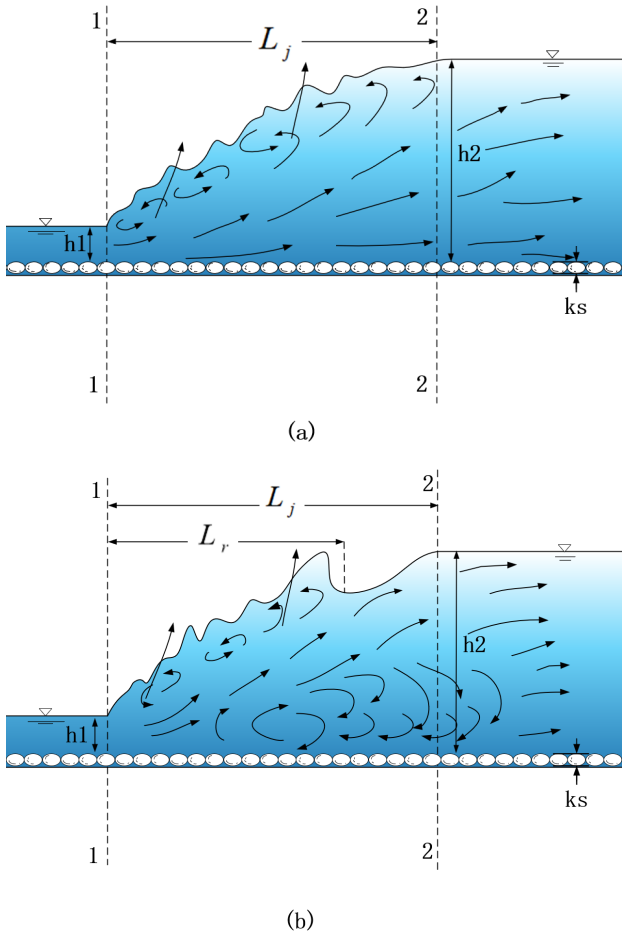


FIGURE 2. Rough bed (a) Developed hydraulic jump (b) Non-Developed hydraulic jump.

riverbed bottom. The rapid flow experiences more intense surface jumping when entering the section before the jump, the length of the hydraulic jump is significantly shortened, and noticeable surface waves are generated in the tailwater section. The structures of both types are shown in Figure 1.

In Figure 1, the cross-section I-I is called the initial cross-section of hydraulic jump, and its water depth h1 at this section is called the initial depth of hydraulic jump. The water section II-II is called the sequent cross-section of hydraulic jump, and its water depth h2 at this section is called the sequent depth of hydraulic jump. The ratio of the sequent depth of hydraulic jump to the initial depth of hydraulic jump is known as the sequent depth ratio. The difference between the initial depth of hydraulic jump and the sequent depth of hydraulic jump is known as the jump height, and the distance between the initial cross-section and the sequent cross-section is called the jump length, denoted by the L_j. As shown in Figure 1(b), L_r (i.e., the roller length) is the horizontal distance between the end of the roller and the toe. Furthermore, for a developed hydraulic jump, the values of L_r is consistent with L_j [35]. Additionally, rough riverbed also has a significant impact on the characteristics of the hydraulic jump. Its structure is shown in Figure 2.

Rough bed can effectively reduce the length and tailwater depth of hydraulic jumps. Despite the complex water flow dynamics in the jump region, numerous empirical formulas based on experimental data [10], [36], [37], [38], [39]have been used to estimate jump length. With advancements in measurement technologies, research on hydraulic jumps has made significant progress on multiple levels, not only including the macroscopic characteristics of hydraulic jumps but also encompassing the microscopic hydraulic phenomena within them. These studies provide important references for understanding the mechanisms of hydraulic jumps and improving the design of energy dissipators.

B. HYDRAULIC JUMP ON ROUGH BED

The main factors influencing hydraulic jumps on rough riverbeds are affected by the hydraulic conditions of water flow, the roughness dimensions of the riverbed, and the characteristics of the fluid. Among them, the length of the

hydraulic jump on the riverbed (L_r) is determined by a series of parameters, including the acceleration due to gravity (g), roughness height (ks), the initial depth of hydraulic jump, the sequent depth of hydraulic jump, upstream flow velocity, and the dynamic viscosity of the fluid. These parameters collectively reveal the complex interactions between the characteristics of the hydraulic jump and the roughness of the bed.

$$L_r = f\ (k_s, g, h_1, h_2, v_1, v) \tag{1}$$

By applying the principles of dimensional analysis (*Buckingham* π *theorem*), the following equation can be obtained:

$$\frac{L_r}{h_1} = f\left(\frac{k_s}{h_1}, \frac{v_1 h_1}{v}, \frac{h_2}{h_1}, \frac{V_1}{\sqrt{g h_1}}\right) \tag{2}$$

In which, $\frac{V_1}{\sqrt{g h_1}}$represents the Froude number at the starting position of the water jump, and $\frac{v_1 h_1}{v}$ corresponds to the Reynolds number approaching the flow rate. Since the value of the Reynolds number is large, the effects of viscosity can be considered negligible. Finally, the following equation can be derived:

$$\frac{L_r}{h_1} = f\left(\frac{k_s}{h_1}, \frac{h_2}{h_1}, F_{r1}\right) \tag{3}$$

C. DATA

This paper uses the experimental dataset provided by Carollo et al. [40] to assess the precision of various deep learning models in estimating the length of hydraulic jumps and their stages of development. The experiments were executed in a rectangular flume of 14.4 meters in length, 0.6 meters in width, and 0.6 meters in depth.

The experimental tank featured walls and a base made of glass, rendering the base into a smooth bed itself. For simulating a rough base, coarse gravel was directly laid and secured. The entire experiment commenced from a point situated 3.5 meters downstream from the inlet section, extending as far as 3 meters in its measuring phase. Figure 3 provides a graphical overview of the experiment.

The rough base consisted of gravel particles compactly stacked and concreted at the bottom, with a total of five variants of the rough base tested. Each gravel bed's particle size distribution was obtained through a sample of 100 particles, with its characteristic value of d_{50} standing at 0.46 cm, 0.82 cm, 1.46 cm, 2.39 cm, and 3.20 cm, respectively. The term d_{50} in this experiment refers to the diameter under which 50% of the particles fall, this metric was employed to characterize the height of roughness, denoted as (ks).

The dataset encompasses a total number of 408 experiments involving the rough base. Of these, 367 sets of data were deemed valid, wherein 196 were complete hydraulic jumps, and 172 were incomplete. The dataset was split in a way that 75% of the data was used for training, while the remaining 25% was set aside for testing. Table 1 presents the change in range of the pre-jump depth, the post-jump depth, the Froude number, the bed roughness' height, and the length of the hydraulic jump in the experimental dataset employed in developing the artificial intelligence models.

TABLE 1. The variation of experimental database used in this study.

Type	ks(cm)	h_1(cm)	h_2(cm)	Fr	L_j(cm)
Train phase	0-3.2	1.11-7.09	8.98-22.06	1.87-9.89	18-85
Test phase	0-3.2	1.34-6.53	9.18-23.45	2.03-9.37	24-90

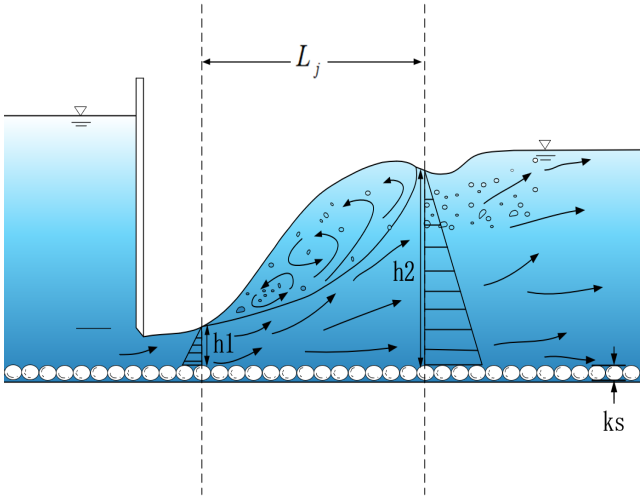


FIGURE 3. Schematic diagram of the experiment.

D. TECHNICAL FLOWCHART

This paper employs CNNs, DNNs, Transformer, and PINNs models to evaluate the length of developed and no-developed hydraulic jumps. The chosen gradient optimizer is the Adam optimizer, and the activation function is the Tanh function. To assess the accuracy of the predicted outputs, the results of selected evaluation metrics are summarized in Table 2. Based on error measurement values, two prior-knowledge-embedded PINN models, named *PINNs*[1] and *PINNs*[2], are identified as the optimal deep learning models for prediction. The classification results of hydraulic jump types using the PINN models are compared with those from other models proposed in this study. The technical flowchart of this research is shown in Figure 4.

1) CONVOLUTIONAL NEURAL NETWORKS (CNNs)

Convolutional Neural Networks (CNNs) are a type of deep neural network designed for grid topology data, such as images. Unlike traditional fully connected networks, CNNs maintain the spatial structure of data through convolution operations, adaptively learn spatial features, and abstract these features across multiple layers, thus

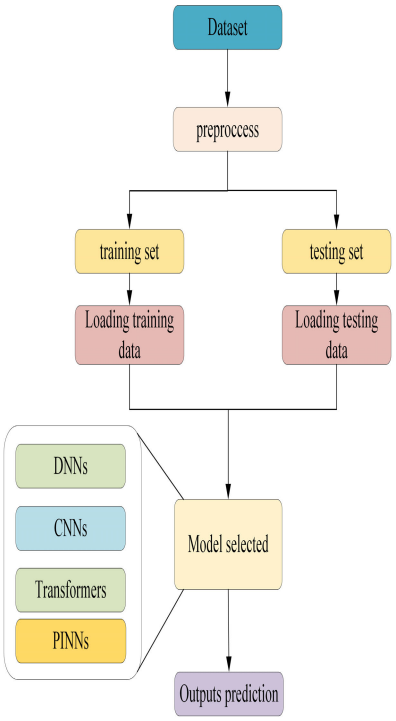


FIGURE 4. Technical Flowchart.

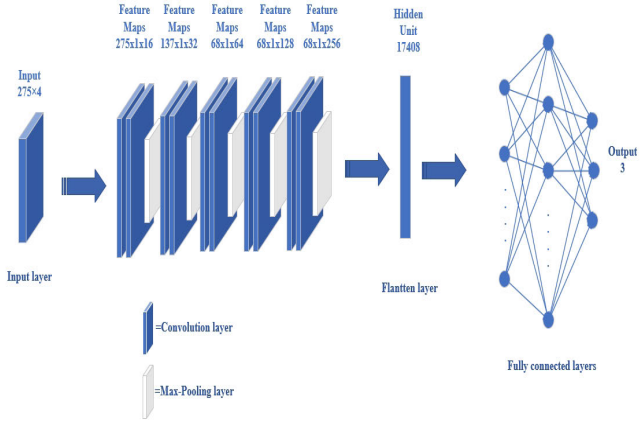


FIGURE 5. The schematic diagram of the CNN model structure.

achieving high-level semantic understanding of the input data. This ability to automatically learn hierarchical feature representations makes CNNs extremely effective in handling spatially structured data like images, establishing them as a mainstream model in image analysis.

Additionally, the application of CNNs has expanded to areas such as computational fluid dynamics, demonstrating significant potential in analyzing numerical simulation flow parameters and predicting changes in flow fields. CNNs can also be used to recover and reconstruct accurate flow datasets from incomplete data, which is crucial for improving the completeness and reliability of flow field testing. Therefore, the advantages of CNNs in processing structured data have prompted many researchers to explore their potential further.

In this research, tabular data is initially transformed into a ''pseudo-image'' format by the CNNs model, which encodes spatial relationships within the data into pixel values to harness the spatial feature extraction capabilities of CNNs. The convolutional neural network architecture employed comprises five convolutional layers along with corresponding pooling layers, which are adjusted after each convolution to either maintain or reduce the dimensionality of the data. The input for prediction is composed of 275 experimental data sets, each represented by four variables. Thus, the CNNs begin with a 1-channel 275×4 matrix as the input. The convolutional layers progressively increase the number of channels (from 16 to 256) and utilize convolutional kernels and pooling layers of varying sizes and strides to either maintain or reduce the length of the data. Finally, a fully connected layer reduces the number of output channels from 256 to 3, which is used to generate the values for regression tasks and the probabilities for classification types.

In the process of performance testing of CNNs model, it was found that the optimization process is constrained by the number of epochs and the learning rate. In order to comprehensively consider obtaining the best results on both the training set and the test set, adjustments were made to the epochs and the learning rate. The technical parameters of the BP algorithm are set as follows: learning rate = 5e-4, initial maximum number of epochs = 40000. The schematic diagram of the CNN model structure used in this article is shown in Figure 5.

2) DEEP NEURAL NETWORKS (DNNs)

Deep Neural Networks (DNNs) are a mainstream and widely used type of neural network. Their distinctive feature is the use of multiple layers of neurons between the network's input and output, allowing them to discover and predict extremely complex associations given a sufficiently large dataset. This characteristic makes deep neural networks versatile, enabling their application across a broad range of tasks including classification, regression, and prediction.

In this research, the DNNs model features an input layer, five fully connected layers, an output layer, and a normalization layer. Initially, the four input features are processed by the first fully connected layer, which contains 64 neurons, to extract preliminary feature information. Subsequently, four hidden layers with the same number of neurons further enhance the extraction and processing of features. Finally, these features are transformed into three outputs through the sixth layer. After processing through a Softmax layer, results suitable for multitasking are obtained.

The technical parameters using the BP algorithm are set as follows: learning rate = 5e-4, initial maximum number of epochs = 20,000. The DNNs model used in this article is shown in Figure 6:

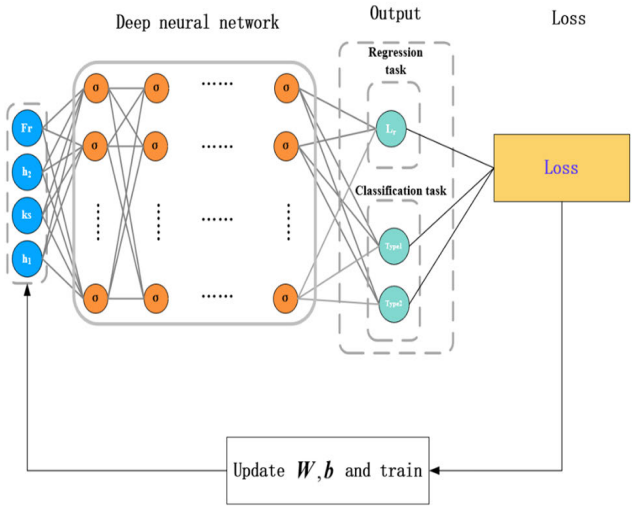


FIGURE 6. Diagram of the DNNs Model.

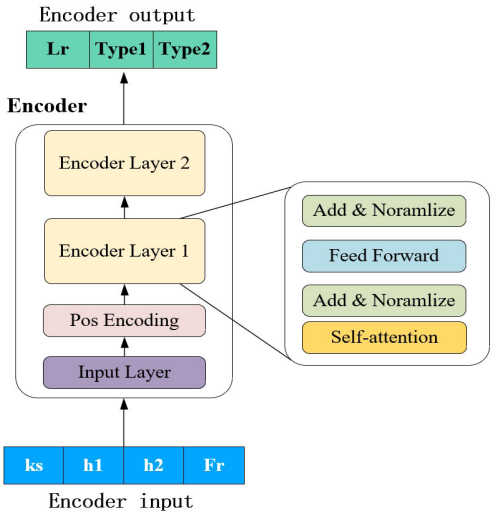


FIGURE 7. Diagram of the Transformer Model.

3) TRANSFORMER MODEL (TRANSFORMER)

The Transformer is a sequence model based on attention mechanisms, initially used in natural language processing. In recent years, due to its excellent ability to capture long-term dependencies and efficient parallel computation, the Transformer has begun to be applied in the field of fluid dynamics. This model consists of an encoder, a decoder, and attention mechanisms, allowing for better modeling of long-term dependency relationships. Compared to traditional convolutional neural networks, the Transformer is sensitive to the order of data and better utilizes contextual associations between sequences, providing unique advantages in capturing the spatiotemporal evolution patterns of dynamic flow fields.

This study does not involve temporal information, thus there is no need to generate sequences. Instead, the task is to extract useful features from the input data for classification or regression purposes. Consequently, the Transformer model employed in this research initially maps the input data dimensionality from 4 to 64 through a linear layer. Subsequently, positional encoding is applied to enhance the representation of position information within the sequence. The core of the model consists of several Transformer encoder layers, each featuring a dimensionality of 64, eight attention heads, and a feed-forward network with a dimensionality of 2048. The aim is for the Transformer model to capture complex dependencies within the input data. Additionally, the model utilizes a dropout rate of 0.1 to mitigate the risk of overfitting, and the ReLU activation function to enhance training stability and non-linear expression capabilities. Ultimately, the output from the Transformer encoder is mapped onto numerical and categorical predictions through another linear layer, and normalized via a Softmax layer to produce the final category probability distribution. The technical parameters for the backpropagation algorithm are set as follows: learning rate = 5e-4, initial maximum number of epochs = 20,000. The Transformer structure used is shown in Figure 7:

4) PHYSICS-INFORMED NEURAL NETWORKS

Although machine learning methods have shown potential and achieved initial success in practice, they often fail to extract interpretable information from large amounts of data. Pure data-driven models might fit observational results well, but due to extrapolation or observational biases, predictions can be physically inconsistent or unreliable [41], [42]. Therefore, it is necessary to combine machine learning models with fundamental physical laws and domain knowledge. Algorithms such as ''Physics-Informed Neural Networks'' (PINNs) utilize prior knowledge to improve the interpretability and predictive accuracy of models [42]. This approach is particularly suitable for dealing with complex systems and long-distance spatiotemporal interactions, such as turbulence and viscoelastic materials. Using PINNs can produce more interpretable methods even in the presence of imperfect data (such as missing or noisy values, outliers, etc.), providing precise and physically consistent predictions, and are even applicable for extrapolation or generalization tasks [43].

The core idea of PINN (Physics-Informed Neural Networks) is to use neural networks to fit given data points while satisfying known physical constraints, integrating data and prior knowledge [44], [45], [46], [47] or physical laws [48], [49], [50]. By interpreting the connections between input and output variables, PINN deepens our understanding of the physical world.

In this study, we incorporate two physical equations into Physics-Informed Neural Networks (PINNs) based on the framework of Deep Neural Networks (DNNs). The PINN algorithm employed is specifically designed for solving straightforward forward problems, integrating prior knowledge during the training process. The network architecture

is illustrated in Figure 8. The prior knowledge utilized includes:

1. **The Wu Chigong Formula** : Derived from experimental data fitting, the Wu Chigong formula [8] is employed to calculate the length of hydraulic jumps. This formula is widely used in practical applications due to its high reliability. The equations are given as follows:

$$L_j = 10 \cdot (h2 - h1)Fr^{-0.32} \tag{4}$$

2. **Fu Minghuan et al**.**Optimized Formula** : Fu Minghuan and colleagues [9] optimized the classical hydraulic theory-based formula for calculating the length of hydraulic jumps. This optimized formula considers various influencing factors, significantly enhancing the calculation accuracy, with the final fitting error being only 4%. The equation is presented as follows:

$$L_j = 0.9782 \cdot (Fr^{1.5} - 1)^2(Fr^{1.001} - 1)^{-2.5035}(h2 + h1) - ks \tag{5}$$

In the formula, F_r, h_1, h_2, k_s *and* L_j represent the Froude number, the initial depth of hydraulic jump,the sequent depth of hydraulic jump, the roughness height, and the length of the hydraulic jump, respectively.

By embedding these physical formulas into the PINNs, we aim to leverage the strengths of both data-driven and physics-based approaches to improve the accuracy and robustness of hydraulic jump length predictions. This methodology ensures that the model adheres to known physical laws while learning from empirical data, leading to more reliable and generalizable predictions in hydraulic engineering applications.

The loss function is a key component in the training of the PINN algorithm, and this paper considers two aspects of loss:

1. Data Retention Loss

This loss function measures the discrepancy between the network's predicted results and the actual observational data, ensuring the model has high predictive accuracy at known data points. By minimizing this loss, DNNs can effectively fit and retain information from the training data.

2. Equation Loss

This loss function measures how well the DNNs' predictions satisfy the governing equations, reflecting the constraints imposed by physical laws. By minimizing this loss, the DNNs' predictions will more closely adhere to the physical laws and rules modeled.

The organic combination of these two loss components allows the PINN algorithm to learn both data and physical knowledge simultaneously, fitting observational data to the maximum extent while ensuring physical consistency, thereby obtaining a more accurate and reasonable prediction model. The design of the loss function reflects the core idea of the PINN method—based on data, constrained by physical laws, to achieve effective integration.

In the standard framework of Physics-Informed Neural Networks (PINNs), a deep neural network (DNN) with multiple hidden layers, denoted as $L_r(F_{r_i}, h_{1_i}, h_{2_i}, k_{s_i}, \theta)$, is used to approximate the solutions to equations, represented by $\hat{L}_j$. This network incorporates trainable weights w and biases b, indicated by θ, and uses a nonlinear activation function, also referred to as σ. Inputs to the DNN include the Froude number(F_r), the initial depth of hydraulic jump(h_1), the sequent depth of hydraulic jump(h_2), and roughness height(ks). The output of the network, which is the solution to the equation, is also denoted as $L_r(F_r, h_1, h_2, k_s)$. This setup defines the general formulation of the equation within the PINNs framework.

$$L_r(F_r, h_1, h_2, k_s) + \mathcal{N}[L_r] = 0 \tag{6}$$

In this context, F_r, h_1, h_2, k_srefers to the input vector, while $\mathcal{N}[\cdot]$ represents the nonlinear operator. Thus, the residual of the equation can be expressed as:

$$e := L_r(F_r, h_1, h_2, k_s) + \mathcal{N}[L_r] \tag{7}$$

The observational dataset is denoted as $\{L_{r_i}, F_{r_i}, h_{1_i}, h_{2_i}, k_{s_i}\}$, consisting of known data points from experimental measurements. These data points will be used to train the network, ensuring that the network can accurately fit the known conditions.

The neural network architecture includes an input layer that receives the observational dataset, followed by a series of hidden layers that perform nonlinear transformations on the input data. Each hidden layer applies a mapping transformation to the output of the previous layer, specifically using a hyperbolic tangent (tanh) nonlinear activation function.

Let (y_k) denote the hidden variable of the k-th hidden layer. The neural network can thus be described as follows:

$$\begin{aligned} lly^0 &= (Fr, h_1, h_2, k), \\ y^k &= \sigma(W^k y^{k-1} + b^k), \quad 1 \le k \le L-1 \end{aligned} \tag{8}$$

In this configuration, (W_k) and (b_k)respectively signify the weight matrix and bias vector of the k-th layer, (σ) indicates the chosen nonlinear activation function. The hidden variable (y^0) represents the input data.

Finally, the output layer generates an approximate solution$(\hat{y})$by applying a linear transformation to the output of the last hidden layer.

$$\hat{y} = W^k y^{k-1} + b^k, \quad k = L \tag{9}$$

Following this, the residual point set $\{F_{r_i}, h_{1_i}, h_{2_i}, k_{s_i}\}$is defined to enforce the network's compliance with the control equations.

By computing the residuals within equations (12), (13), or (14) — specifically, the differences between the left and right sides of these equations — we construct the loss terms for each equation. Subsequently, the losses from data retention and the physical equations are combined through weighted summation to derive the comprehensive total loss function, denoted as loss $\mathcal{L}$. The configuration of

weights demonstrates the emphasis placed on different loss components.

$$\mathcal{L} = w_{data}\mathcal{L}_{data} + w_{eq}\mathcal{L}_{eq} \tag{10}$$

In which:

$$ll\mathcal{L}_{data} = \frac{1}{N_{data}}\sum_{i=1}^{N_{data}}(y_i - \hat{y}_i)^2$$

$$\mathcal{L}_{eq} = \frac{1}{N_{eq}}\sum_{i=1}^{N_{eq}}(e)^2 \tag{11}$$

The weights w_{data} and w_{eq} are used to balance the interaction between the two types of losses. These weights can be defined by the user or adjusted automatically and play a crucial role in enhancing the trainability of Physics-Informed Neural Networks (PINNs) [51], [52]. In this context, y_i represents the true data values, $\hat{y}_i$ denotes the predicted data values, and e symbolizes the residuals of the equations.The technical parameters of the BP algorithm are set as follows: learning rate = 5e-4, initial maximum number of epochs = 25000. The PINN algorithm is illustrated in Figure 8.

5) PERFORMANCE METRICS FOR MODEL EVALUATION

This paper utilizes three different standards: Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and the Coefficient of Determination (R^2), to evaluate the performance of regression tasks in applied models. Additionally, it uses recall, and precision to assess the performance of classification tasks.

The method for calculating RMSE is outlined as follows:

$$RMSE = \sqrt{\frac{1}{N}\sum_{j=1}^{N}(L_{j_o} - L_{j_p})^2} \tag{12}$$

In the formula, N is the number of data points in the dataset, L_{j_o} and L_{j_p} respectively represent the observed and simulated values of the hydraulic jump length.

The calculation method for MAE is as follows:

$$MAE = \frac{1}{N}\sum_{i=1}^{n}|L_{j_o} - L_{j_p}| \tag{13}$$

In the formula, N is the number of data points in the dataset, and respectively represent the observed and simulated values of the hydraulic jump length. This formula directly reflects the average size of the prediction error through the absolute error.

The method for calculating R^2 is outlined as follows:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(L_{j_o} - L_{j_p})^2}{\sum_{i=1}^{n}(L_{j_o} - \bar{L}_{j_o})^2} \tag{14}$$

where L_{j_o} is the j-th actual observed value, L_{j_p} is the j-th predicted value, and is the average of the actual observed values. The value of R2 ranges from 0 to 1, with values closer to 1 indicating a better fit of the model.

The calculation method for Precision is as follows:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{15}$$

In the formula, TP (True Positives) refers to the number of positive instances correctly predicted, and FP (False Positives) refers to the number of negative instances incorrectly predicted as positive. Precision measures the proportion of actual positive cases among the samples predicted as positive.

The calculation method for Recall is as follows:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{16}$$

In this context, FN (False Negatives) represents the number of positive instances incorrectly predicted as negative. Recall assesses the percentage of actual positive samples that are accurately identified as positive.

III. RESULTS AND DISCUSSION

The study utilizes the Froude numbers, the roughness heights, the initial depth of hydraulic jump,the sequent depth of hydraulic jump data as inputs for the model. Through the implementation of Physics-Informed Neural Networks (PINNs) and three other data-driven neural network models, the research aims to predict the length and classify the hydraulic jumps on horizontal river beds. The findings demonstrate that the PINNs algorithm achieves superior accuracy in both regression and classification tasks. Regarding time efficiency, the PINNs approach exhibits slightly faster performance compared to the data-driven models. The results of the four models, including PINNs employing two different algorithms, are summarized in Table 2.

A. CONVOLUTIONAL NEURAL NETWORKS (CNNS) MODEL

As shown in Table 2, the Convolutional Neural Network (CNN) model demonstrates a high prediction accuracy of 98.55% on the training dataset, with an equally high recall rate of 98.55%. These values indicate the model's efficacy in learning from the training data and correctly identifying most instances. The coefficient of determination (R^2) of 91.10% further supports this performance, signifying that a substantial proportion of the variance in hydraulic jump lengths is explained by the model. The Mean Absolute Error (MAE) stands at 2.5264(cm), and the Root Mean Squared Error (RMSE) is 3.5731(cm). The relatively low MAE suggests that, on average, the model's predictions are very close to the actual values. The RMSE, which penalizes larger errors more than the MAE, also remains reasonably low, indicating that large prediction errors are infrequent.

On the testing dataset, the CNN model maintains strong performance, achieving a prediction accuracy of 95.72% and a recall rate of 95.65%. As expected, there is a slight decrease in performance due to the model's exposure to new, unseen data. However, the high accuracy and recall rates

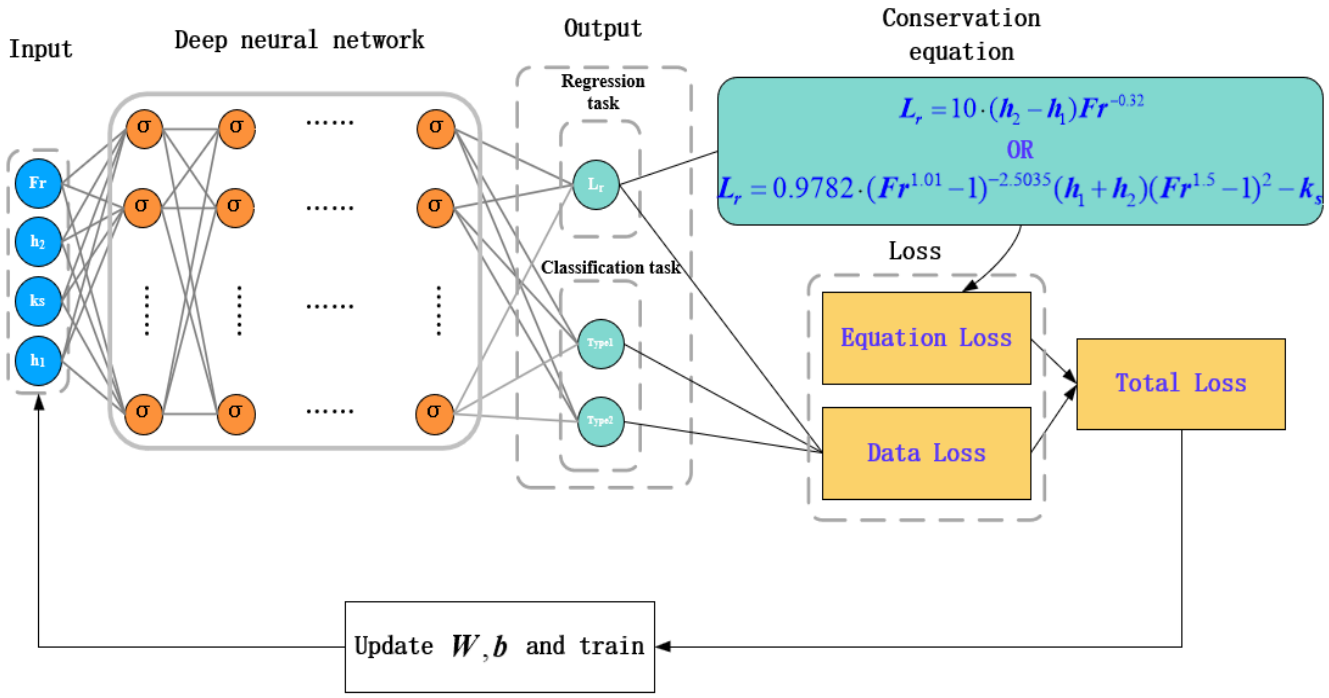


FIGURE 8. Diagram of the PINNs Model.

TABLE 2. Miti-model simulation results for developed and non-developed hydraulic jumps.

Pattern No.	Model	Train phase					Test phase				
		R^2	*MAE*(cm)	*RMSE*(cm)	Precision	Recall	R^2	*MAE* (cm)	*RMSE*(cm)	Precision	Recall
1	CNNs	0.9110	2.5264	3.5731	0.9855	0.9855	0.8728	3.4443	4.6295	0.9572	0.9565
2	DNNs	0.9300	2.4032	3.1688	0.9964	0.9964	0.8458	3.7764	5.0981	0.9565	0.9565
3	Transformer	0.9481	1.5382	2.7283	0.9387	0.9382	-0.3020	11.8339	14.8122	0.4683	0.4683
4	*PINNs*[1]	0.9496	1.9305	2.6892	0.9964	0.9964	0.8638	3.5535	4.7905	0.9783	0.9783
5	*PINNs*[2]	0.9416	2.1109	2.8932	0.9856	0.9855	0.8818	3.3864	4.4627	0.9677	0.9674

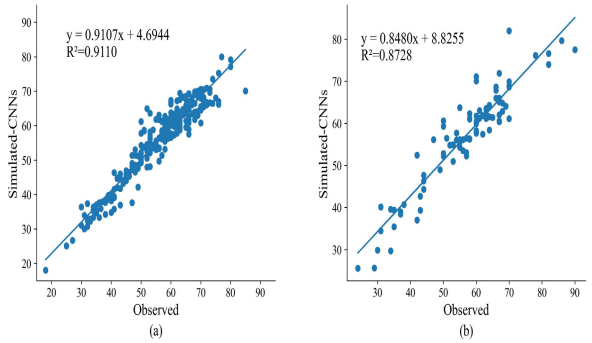


FIGURE 9. The scatterplot of observed and simulated hydraulic jump length using CNNs for training (a) and testing (b) period.

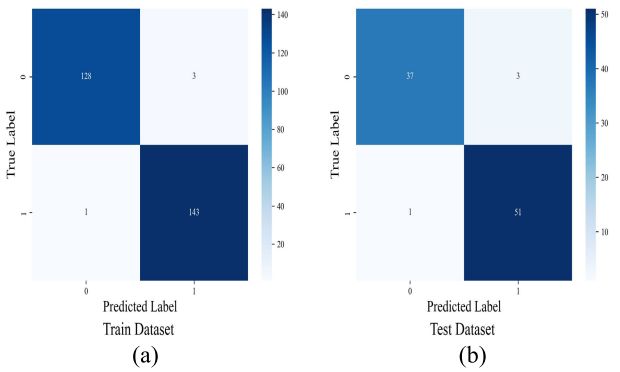


FIGURE 10. The Confusion Matrix of observed and simulated hydraulic jump length using CNNs for training (a) and testing (b) period.

demonstrate the model's generalization capability beyond the training data. The testing dataset's R^2 value of 0.8728, although slightly lower than the training R^2, still indicates a strong correlation between the predicted and actual values. The MAE of 3.4443(cm) and RMSE of 4.6295(cm), while higher than the training errors, remain within an acceptable range, reinforcing the model's robustness and reliability in making accurate predictions.

Figure 9 illustrates the scatter plot of observed versus predicted hydraulic jump lengths using the optimal CNN model. The proximity of the data points to the line of

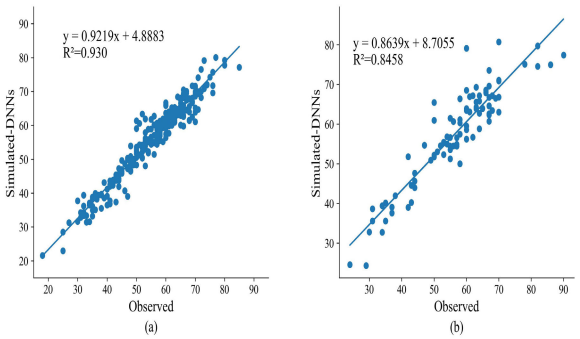


FIGURE 11. The scatterplot of observed and simulated hydraulic jump length using DNNs for training (a) and testing (b) period.

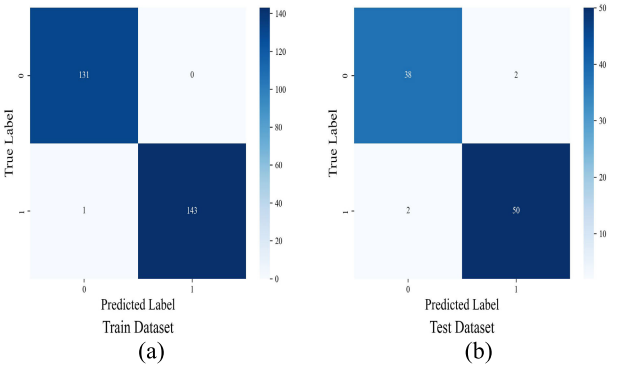


FIGURE 12. The Confusion Matrix of observed and simulated hydraulic jump length using DNNs for training (a) and testing (b) period.

perfect prediction (where observed equals predicted) visually confirms the model's accuracy. Most data points cluster closely around this line, indicating precise predictions.

Figure 10 presents the confusion matrix, providing a clear visualization of the classification results. Each column of the matrix represents instances within a predicted category, while each row represents instances within an actual category. All correct predictions are located on the diagonal, making any non-zero values off the diagonal easily identifiable as errors, thus allowing for visualization of mistakes. The results indicate that CNNs can serve as a reliable alternative to empirical equations for simulating hydraulic jump lengths and performing classification tasks.

B. DEEP LEARNING NEURAL NETWORKS (DNNS) MODEL

As shown in Table 2, the Deep Neural Network (DNN) model achieved a prediction accuracy and recall rate of 99.64% on the training dataset, with a coefficient of determination (R^2) of 93.00%. The Mean Absolute Error (MAE) was 2.4032(cm), and the Root Mean Squared Error (RMSE) was 3.1688(cm). On the test dataset, the model's prediction accuracy was 95.65%, recall rate was 95.65%, and R^2 was 0.8458. The MAE was 3.7764(cm), and the RMSE was 5.0981(cm). Figures 11 and 12 provide scatter plots of observed versus simulated lengths of developed and no-developed hydraulic jumps, as well as classification graphs for type prediction based on optimal results. The results demonstrate that DNNs perform well in regression and classification tasks for hydraulic jumps, showing better fitting on the test set compared to CNNs, although their generalization performance is slightly inferior to CNNs.

C. TRANSFORMERS MODEL

Various metrics of the model are shown in Table 2.the Transformers Model demonstrated high accuracy, precision, and efficiency on the training set with notable metrics such as an accuracy of 93.87%, a recall rate of 93.82%, and a coefficient of determination (R^2) of 0.9481.The Mean Absolute Error (MAE) was recorded at 1.5382(cm), with a root mean squared error (RMSE) of 2.7283(cm). However,the

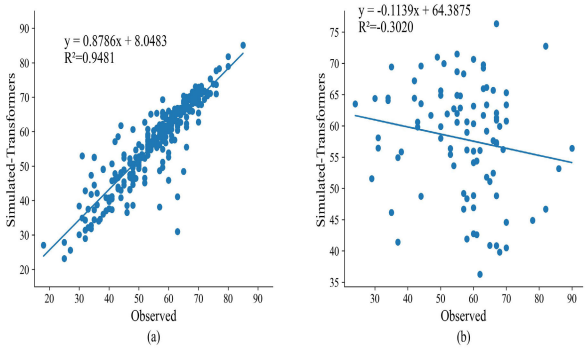


FIGURE 13. The scatterplot of observed and simulated hydraulic jump length using Transformer for training (a) and testing (b) period.

performance significantly differed in the testing dataset, showing an equal accuracy and recall rate of 46.83%, a negative R^2 of −0.3020, and substantially higher MAE and RMSE values of 11.8339(cm) and 14.8122(cm), respectively. These results highlight the challenges in model generalization beyond the training dataset. Figures 13 and 14 illustrate the scatter plots of observed and simulated hydraulic jump lengths, and the classification chart of type predictions, respectively. The research results indicate that the model fits well on the training set data in terms of hydraulic jump prediction. However, its lack of generalization becomes evident when tested, and its performance in classification tasks lags significantly behind DNNs and CNNs. Therefore, from the perspective of this research, the Transformer model is not suitable for such multi-task prediction.

D. PHYSICS-INFORMED NEURAL NETWORKS MODEL

Based on the framework of DNNs, two physical equations were used for knowledge embedding in PINNs. Firstly, the Wu et al.'s formula [8] was employed, which is renowned for its simplicity in computation and widespread use in practical engineering applications and is considered to have good generalization capabilities across multiple scenarios. The research results show that in the prediction of hydraulic jump lengths when using the Wu Chihong formula as the basis for knowledge embedding, the model achieved a Mean Absolute

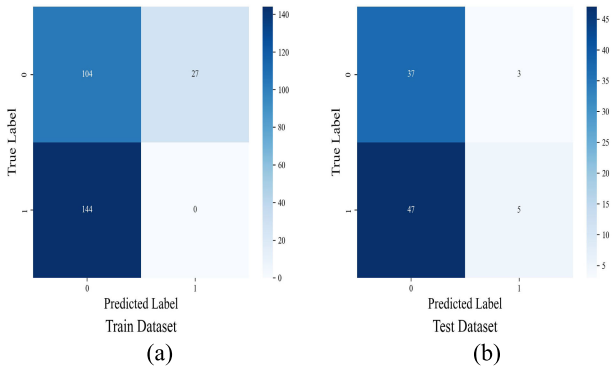


FIGURE 14. The Confusion Matrix of observed and simulated hydraulic jump length using Transformer for training (a) and testing (b) period.

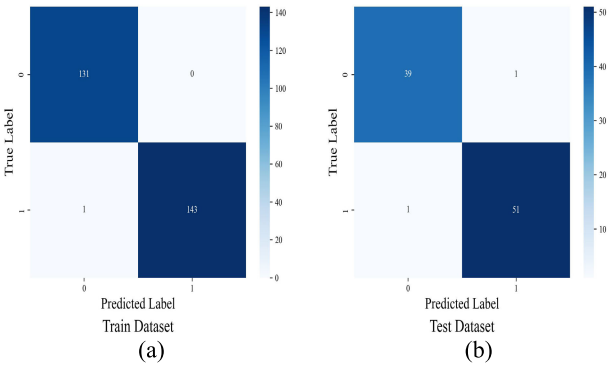


FIGURE 16. The Confusion Matrix of observed and simulated hydraulic jump length using *PINNs*[1] for training (a) and testing (b) period.

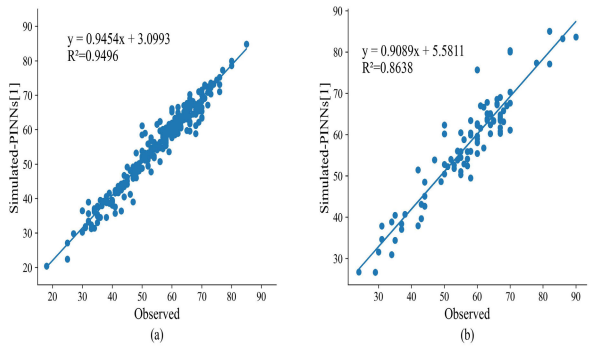


FIGURE 15. The scatterplot of observed and simulated hydraulic jump length using *PINNs*[1] for training (a) and testing (b) period.

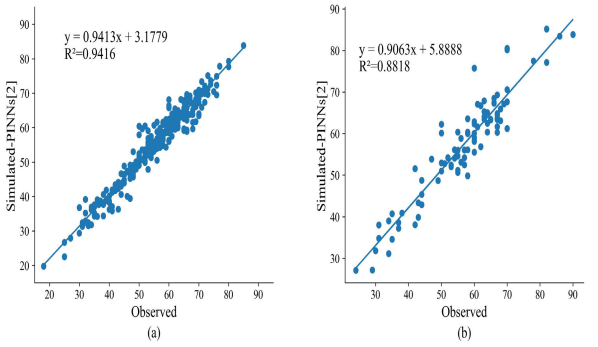


FIGURE 17. The scatterplot of observed and simulated hydraulic jump length using *PINNs*[2] for training (a) and testing (b) period.

Error (MAE) of 1.9305(cm) and a Root Mean Squared Error (RMSE) of 2.6892(cm) on the training set; on the test set, the MAE was 3.5535(cm) and the RMSE was 4.7905(cm). As shown in Figure n, the linear relationship between the observed and predicted data during the training and testing phases reached correlation coefficients of 0.9496 and 0.8638, respectively, indicating the model's high linear fitting capability to the data. Figure n also displays the classification of samples using this algorithm, where all accurate predictions are located on the diagonal of the confusion matrix, demonstrating its exceptional accuracy in classification performance.

The second method involves embedding a semi-theoretical and semi-empirical formula derived by Fu Minghuan et al. [9]. through the analysis of experimental data. This formula accurately reflects the general laws of hydraulic jump lengths. Subsequently, the formula is embedded into the Physics-Informed Neural Network (*PINNs*[2]), allowing the network to consider the constraints of physical laws while learning from the data, thereby improving the model's prediction accuracy and generalization ability. The results indicate that when predicting hydraulic jump lengths using this method, the MAE and RMSE obtained on the training set are 2.1109(cm) and 2.8932(cm), respectively, while on the test set, they are 3.3864(cm) and 4.4627(cm), respectively. Figure 14 and Figure 16 reveal that the linear relationship of the data on the training and test sets shows correlation coefficients of 0.95 and 0.88, demonstrating high linearity in data fitting. Further analysis suggests that the second formula performs better in approximating the real situation of hydraulic jumps. In terms of classification ability, the confusion matrix shown in Figure 15 and Figure 17 demonstrates that the second method has similar outstanding classification performance compared to the first method.

These findings highlight the effectiveness of both PINN methods in predicting hydraulic jump lengths and provide valuable references for future engineering applications.

After analyzing the four models and two algorithms, the results show that the models using two different algorithms for PINN outperform other models in predicting the lengths of both complete and incomplete hydraulic jumps. Compared to the regression task in reference [13], we have achieved similarly good results in our regression predictions. However, compared to the classification task in reference [53], our results in classification tasks are significantly better. These findings demonstrate that our model maintains a high level of effectiveness and accuracy when employing a multi-task learning strategy. Moreover, comparing the data, although the performance of the Transformer model is relatively average, Convolutional Neural Networks (CNNs), Deep

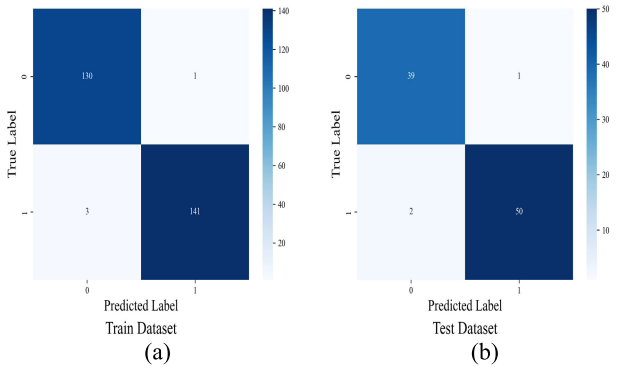


FIGURE 18. The Confusion Matrix of observed and simulated hydraulic jump length using *PINNs*[2] for training (a) and testing (b) period.

Neural Networks (DNNs), and Physics-Informed Neural Networks (PINNs) all demonstrate their potential for multi-tasking. More importantly, the *PINNs*[1] and *PINNs*[2] models achieved R^2 values of 0.8638 and 0.8818, respectively, in the test phase, both higher than those of the CNN and DNN models. This indicates that physics-informed neural networks have significant potential in handling complex fluid dynamics problems. Additionally, in terms of MAE and RMSE metrics, *PINNs*[1] and *PINNs*[2] also exhibited excellent performance, indicating lower prediction errors. Notably, the *PINNs*[2] model achieved the lowest values in the test phase, with an MAE of 3.3864(cm) and an RMSE of 4.4627(cm).

IV. CONCLUSION

This study explores the application of four neural network models—DNNs, CNNs, Transformers, and PINNs—and two PINN knowledge integration algorithms in predicting characteristics of hydraulic jumps. The proposed AI models were trained and tested using the Froude numbers, the roughness heights, the initial depth of hydraulic jump, the sequent depth of hydraulic jump data to perform multi-task predictions of hydraulic jump length and its development stages. By employing a multi-task learning framework, the study not only reduces the total number of parameters compared to training two separate models but also improves training efficiency. Furthermore, shared feature representations enhance the model's generalization capability across different tasks. After thoroughly comparing the four models, the results indicate that PINNs models, by directly incorporating physical knowledge, enhance the physical interpretability and accuracy of the models. This approach considers the constraints of physical laws, effectively improving the prediction accuracy and generalization capability of the models. This method is particularly suitable for scenarios with limited or low-quality data, as it relies on known physical principles to guide the learning process, reducing the dependence on large-scale data. From the training set to the test set, the significant performance drop of the Transformer model suggests an overfitting problem, as the model performs well on the training data but fails to generalize to new, unseen data effectively. The negative R^2 during the test phase indicates a substantial discrepancy between the model's predictions and the actual values. In contrast, the Physics-Informed Neural Network (PINN) models, especially *PINNs*[2], demonstrated superior performance compared to traditional neural networks and the Transformer model. The *PINNs*[2] model achieved an accuracy of 0.9677 and a recall of 0.9674 in the test phase, improving the accuracy and recall by approximately $2-3\%$ over the CNN and DNN models. These results further validate the advantage of physics-informed neural networks over traditional deep learning models in the task of predicting hydraulic jump lengths. They highlight the effectiveness of incorporating physics-based constraints into neural networks for modeling complex fluid dynamics problems, such as hydraulic jumps.

In addition, by comparing these different types of models in the prediction of the steady and unsteady stages of hydraulic jumps, it is evident that Physics-Informed Neural Networks (PINNs) may more accurately capture and simulate the variations in hydraulic jump lengths due to their high precision and efficiency in handling complex physical processes. Such comparisons not only help us understand the applicability of each model to specific physical problems but also promote the complementarity and integration between data-driven and physics-driven methods. Applying these models in real-time monitoring systems can enhance the management and control of hydraulic structures, providing timely predictions and alerts.

Despite the promising results, our study has several limitations:

1. Data Quality and Quantity: The dataset used in this study, although comprehensive, may not cover all possible scenarios and conditions of hydraulic jumps. The limited data quantity and variability might restrict the generalizability of the models.

2. Simplified Physical Models: While PINNs incorporate physical laws, the simplifications and assumptions inherent in these models might not capture all the nuances of real-world hydraulic jumps. Further refinement of these physical models is necessary for more accurate simulations.

3. Integration with Real-time Systems: Although we discuss the potential application of these models in real-time monitoring systems, practical implementation and validation in real-world conditions are yet to be explored.

As deep learning algorithms continue to develop and improve, their applications in hydraulic jump research will become more extensive and in-depth. Future research should focus on error analysis, comparison with traditional methods, real-time applications, scalability, and integration with other models to further enhance the effectiveness and practicality of these models in hydraulic engineering. Moreover, hydraulic jump research will provide new application scenarios and challenges for the development of deep learning, fostering the intersection of artificial intelligence technology and hydraulic engineering. This interdisciplinary research not only advances technological progress in the field of hydraulic

engineering but also opens new directions and application prospects for the development of deep learning technology.

APPENDIX

ABBREVIATION LIST

Physics-Informed Neural Networks (PINNs).
Deep Neural Networks (DNNs).
Convolutional Neural Networks (CNNs).
support vector machines (SVM).
Reynolds-averaged Navier-Stokes (RANS).
Mean Square Error (RMSE).
the Coefficient of Determination (R^2).
Mean Absolute Error (MAE).
TP (True Positives).
FP (False Positives).
FN (False Negatives).

REFERENCES

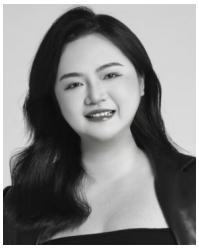
[1] W. H. Hager, *Energy Dissipators and Hydraulic Jump*. Dordrecht, The Netherlands: Kluwer, 1992.
[2] M. Aamir, Z. Ahmad, M. Pandey, M. A. Khan, A. Aldrees, and A. Mohamed, "The effect of rough rigid apron on scour downstream of sluice gates," *Water*, vol. 14, no. 14, p. 2223, Jul. 2022, doi: 10.3390/w14142223.
[3] H. Hamidifar and M. Nasrabadi, "Scour downstream of a rough rigid apron," *World Appl. Sci. J.*, vol. 14, no. 8, pp. 1169–1178, 2011.
[4] S. Farzin and M. V. Anaraki, "Optimal construction of an open channel by considering different conditions and uncertainty: Application of evolutionary methods," *Eng. Optim.*, vol. 53, no. 7, pp. 1173–1191, Jul. 2021.
[5] T. R. Al-Husseini, A. H. Ghawi, and A. H. Ali, "Performance of hydraulic jump rapid mixing for enhancement of turbidity removal from synthetic wastewater: A comparative study," *J. Water Process Eng.*, vol. 30, Aug. 2019, Art. no. 100590.
[6] A. Witt, J. Gulliver, and L. Shen, "Simulating air entrainment and vortex dynamics in a hydraulic jump," *Int. J. Multiphase Flow*, vol. 72, pp. 165–180, Jun. 2015.
[7] H. M. A. Ahmed, M. El Gendy, A. M. H. Mirdan, A. A. M. Ali, and F. S. F. A. Haleem, "Effect of corrugated beds on characteristics of submerged hydraulic jump," *Ain Shams Eng. J.*, vol. 5, no. 4, pp. 1033–1042, Dec. 2014.
[8] C. G. Wu, *Hydraulics (One Volumes)*, vol. 279. Beijing, China: Higher Education Press, 2016.
[9] F. Minghuan and Z. Zhichang, "Calculation of jump length of free hydraulic jump on a roughened bed in a stilling basin," *J. Hydraul. Eng.*, no. 6, pp. 69–75, 2016, doi: 10.16198/j.cnki.1009-640x.2016.06.010.
[10] J. N. Bradley and A. J. Peterka, "The hydraulic design of stilling basins: Hydraulic jump on a horizontal apron (Basin 1)," *J. Hydraulic Division*, vol. 83, no. 5, pp. 1–19, 1957.
[11] S. Nikmehr and Y. Aminpour, "Numerical simulation of hydraulic jump over rough beds," *Periodica Polytechnica Civil Eng.*, vol. 64, no. 2, pp. 396–407, 2020.
[12] D. Velioglu, N. D. Tokyay, and A. I. Dincer, "A numerical and experimental study on the characteristics of hydraulic jumps on rough beds," in *Proc. 36th IAHR World Congr.*, vol. 28, 2015, pp. 1–9.
[13] P. Khosravinia, H. Sanikhani, and C. Abdi, "Predicting hydraulic jump length on rough beds using data-driven models," *J. Rehabil. Civil Eng.*, vol. 6, no. 2, pp. 139–153, 2018.
[14] Y. Liu, X. Zhang, H. Yu, Y. Sun, C. Sun, Z. Li, and X. Li, "Hydraulic model of partial dam break based on sluice gate flow," *Ocean Eng.*, vol. 295, Mar. 2024, Art. no. 116974.
[15] K. Roushangar and F. Homayounfar, "Prediction characteristics of free and submerged hydraulic jumps on horizontal and sloping beds using SVM method," *KSCE J. Civil Eng.*, vol. 23, no. 11, pp. 4696–4709, Nov. 2019.
[16] S. Baharvand, A. Jozaghi, R. Fatahi-Alkouhi, S. Karimzadeh, R. Nasiri, and B. Lashkar-Ara, "Comparative study on the machine learning and regression-based approaches to predict the hydraulic jump sequent depth ratio," *Iranian J. Sci. Technol., Trans. Civil Eng.*, vol. 45, no. 4, pp. 2719–2732, Dec. 2021.
[17] J. Donnelly, S. Abolfathi, J. Pearson, O. Chatrabgoun, and A. Daneshkhah, "Gaussian process emulation of spatio-temporal outputs of a 2D inland flood model," *Water Res.*, vol. 225, Oct. 2022, Art. no. 119100.
[18] F. N. Chianeh, M. V. Anaraki, F. Mahmoudian, and S. Farzin, "A new methodology for the prediction of optimal conditions for dyes' electrochemical removal; Application of copula function, machine learning, deep learning, and multi-objective optimization," *Process Saf. Environ. Protection*, vol. 182, pp. 298–313, Feb. 2024.
[19] J. J. Hopfield and D. W. Tank, "'Neural' computation of decisions in optimization problems," *Biol. Cybern.*, vol. 52, no. 3, pp. 141–152, Jul. 1985.
[20] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, Oct. 1986.
[21] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.
[22] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, Jul. 2006.
[23] X. Guo, W. Li, and F. Iorio, "Convolutional neural networks for steady flow approximation," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2016, pp. 481–490.
[24] S. Ye, Z. Zhang, X. Song, Y. Wang, Y. Chen, and C. Huang, "A flow feature detection method for modeling pressure distribution around a cylinder in non-uniform flows by using a convolutional neural network," *Sci. Rep.*, vol. 10, no. 1, p. 4459, Mar. 2020.
[25] J. Donnelly, A. Daneshkhah, and S. Abolfathi, "Forecasting global climate drivers using Gaussian processes and convolutional autoencoders," *Eng. Appl. Artif. Intell.*, vol. 128, Feb. 2024, Art. no. 107536.
[26] J. Ling, A. Kurzawski, and J. Templeton, "Reynolds averaged turbulence modelling using deep neural networks with embedded invariance," *J. Fluid Mech.*, vol. 807, pp. 155–166, Nov. 2016.
[27] M. Nadda, S. K. Shah, S. Roy, and A. Yadav, "CFD-based deep neural networks (DNN) model for predicting the hydrodynamics of fluidized beds," *Digit. Chem. Eng.*, vol. 8, Sep. 2023, Art. no. 100113.
[28] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," 2017, *arXiv:1706.03762*.
[29] J. Jiang, G. Li, Y. Jiang, L. Zhang, and X. Deng, "TransCFD: A transformer-based decoder for flow field prediction," *Eng. Appl. Artif. Intell.*, vol. 123, Aug. 2023, Art. no. 106340.
[30] B. Xu, Y. Zhou, and X. Bian, "Self-supervised learning based on transformer for flow reconstruction and prediction," *Phys. Fluids*, vol. 36, no. 2, 2024.
[31] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," *J. Comput. Phys.*, vol. 378, pp. 686–707, Feb. 2019.
[32] H. Eivazi, M. Tahani, P. Schlatter, and R. Vinuesa, "Physics-informed neural networks for solving Reynolds-averaged Navier–Stokes equations," *Phys. Fluids*, vol. 34, no. 7, 2022.
[33] J. Donnelly, S. Abolfathi, and A. Daneshkhah, "A physics-informed neural network surrogate model for tidal simulations," in *Proc. ECCOMAS*, 2023, pp. 836–844.
[34] J. Donnelly, A. Daneshkhah, and S. Abolfathi, "Physics-informed neural networks as surrogate models of hydrodynamic simulators," *Sci. Total Environ.*, vol. 912, Feb. 2024, Art. no. 168814.
[35] W. H. Hager, R. Bremen, and N. Kawagoshi, "Classical hydraulic jump: Length of roller," *J. Hydraulic Res.*, vol. 28, no. 5, pp. 591–608, Sep. 1990.
[36] E. A. Elevatorski, *Hydraulic Energy Dissipators*. New York, NY, USA: McGraw-Hill, 1959.
[37] W. H. Hager and R. Bremen, "Classical hydraulic jump: Sequent depths," *J. Hydraulic Res.*, vol. 27, no. 5, pp. 565–585, Sep. 1989.
[38] Z. Qingke and S. Chuanlin, "Experimental study on turbulence characteristics of flow downstream of low froude number hydraulic jump," *J. Hydraul. Eng.*, no. 5, pp. 1–8, 1986.
[39] R. Bremen and W. H. Hager, "T-jump in abruptly expanding channel," *J. Hydraul. Res.*, vol. 31, no. 1, pp. 61–78, 1993.
[40] F. G. Carollo, V. Ferro, and V. Pampalone, "Hydraulic jumps on rough beds," *J. Hydraulic Eng.*, vol. 133, no. 9, pp. 989–999, Sep. 2007.

[41] M. Innes, A. Edelman, K. Fischer, C. Rackauckas, E. Saba, V. B. Shah, and W. Tebbutt, "A differentiable programming system to bridge machine learning and scientific computing," 2019, *arXiv:1907.07587*.
[42] S. Cai, Z. Mao, Z. Wang, M. Yin, and G. E. Karniadakis, "Physics-informed neural networks (PINNs) for fluid mechanics: A review," *Acta Mechanica Sinica*, vol. 37, no. 12, pp. 1727–1738, Dec. 2021.
[43] G. E. Karniadakis, I. G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, and L. Yang, "Physics-informed machine learning," *Nature Rev. Phys.*, vol. 3, no. 6, pp. 422–440, May 2021.
[44] A. Daw, A. Karpatne, W. Watkins, J. Read, and V. Kumar, "Physics-guided neural networks (PGNN): An application in lake temperature modeling," 2017, *arXiv:1710.11431*.
[45] M. Elhamod, J. Bu, C. Singh, M. Redell, A. Ghosh, V. Podolskiy, W.-C. Lee, and A. Karpatne, "CoPhy-PGNN: Learning physics-guided neural networks with competing loss functions for solving eigenvalue problems," *ACM Trans. Intell. Syst. Technol.*, vol. 13, no. 6, pp. 1–23, Dec. 2022.
[46] Y. Zhang, Y. Liu, X. Li, S. Jiang, K. Dixit, X. Zhang, and X. Ji, "PgNN: Physics-guided neural network for Fourier ptychographic microscopy," 2019, *arXiv:1909.08869*.
[47] D. Bertels and P. Willems, "Physics-informed machine learning method for modelling transport of a conservative pollutant in surface water systems," *J. Hydrol.*, vol. 619, Apr. 2023, Art. no. 129354.
[48] L. Lu, X. Meng, Z. Mao, and G. E. Karniadakis, "DeepXDE: A deep learning library for solving differential equations," *SIAM Rev.*, vol. 63, no. 1, pp. 208–228, Jan. 2021.
[49] L. Yang, D. Zhang, and G. E. Karniadakis, "Physics-informed generative adversarial networks for stochastic differential equations," *SIAM J. Sci. Comput.*, vol. 42, no. 1, pp. A292–A317, Jan. 2020.
[50] G. Pang, L. Lu, and G. E. Karniadakis, "FPINNs: Fractional physics-informed neural networks," *SIAM J. Sci. Comput.*, vol. 41, no. 4, pp. A2603–A2626, Jan. 2019.
[51] S. Wang, X. Yu, and P. Perdikaris, "When and why PINNs fail to train: A neural tangent kernel perspective," *J. Comput. Phys.*, vol. 449, Jan. 2022, Art. no. 110768.
[52] S. Wang, Y. Teng, and P. Perdikaris, "Understanding and mitigating gradient flow pathologies in physics-informed neural networks," *SIAM J. Sci. Comput.*, vol. 43, no. 5, pp. A3055–A3081, Jan. 2021.
[53] G. Mahtabi, B. Chaplot, H. M. Azamathulla, and M. Pal, "Classification of hydraulic jump in rough beds," *Water*, vol. 12, no. 8, p. 2249, Aug. 2020.

ZIYUAN XU received the B.S. degree in environmental engineering from Sichuan University of Science and Engineering, Zigong, China. He is currently pursuing the master's degree with Qinghai University. His research interests include hydraulics and river dynamics, physics-informed neural networks, and the application of artificial intelligence in the water field.

ZIRUI LIU received the bachelor's degree in electrical engineering and automation and the master's degree in radio physics from Mudanjiang Teachers College, Mudanjiang, China, in 2016 and 2024, respectively.

Since 2018, he has been the Head of teaching with Hunan Nod Education Technology Company Ltd. His research interests include computer vision, image processing, and deep learning.



YINGZI PENG received the bachelor's and master's degrees in chemical engineering and technology from Central South University, Changsha, China, in 2012 and 2015, respectively.

From 2015 to 2016, she worked as a Quality System Engineer at Hunan Yunifang Cosmetics Company Ltd. Since 2018, she has been the Founder of the company, she nodded in Hunan Education Technology Company Ltd. Her research interests include deep learning and artificial intelligence.