Prediction of Effective Width of Varying Depth Box-Girder Bridges Using Convolutional Neural Networks

Kejian Hu and Xiaoguang Wu

Highway School, Chang’an University, Xi’an, China

Correspondence should be addressed to Xiaoguang Wu; wxgwst.cn@126.com

Received 14 October 2021; Revised 14 December 2021; Accepted 17 December 2021; Published 10 March 2022

Copyright © 2022 Kejian Hu and Xiaoguang Wu. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Effective flange width is widely used in bridge design to consider the effect of shear lag. The simplified formula for the effective flange width of box girder bridges of variable depth in existing codes and studies may not be conservative, and accurate methods, such as the finite element method, are time-consuming. The purpose of this research is to develop a method that uses a convolutional neural network (CNN) to predict the effective width of box girder bridges of varying depths. These models have been trained, validated, and tested on datasets generated from thousands of finite element models. The lower error in the test set indicates that the CNN model can be used to predict the effective width. In addition, the impact of different architectures is also studied. The proposed method makes real-time analysis possible and has a wide range of applications in the analysis and design of box-girder bridges at different depths.

1. Introduction

The shear-lag effect is a phenomenon of uneven distribution of normal stress, which exists in various civil engineering structures, such as box-girder bridges. This phenomenon can lead to erroneous estimations of displacement and stress in extreme fibers based on beam bending theory [1–3]. As the width of the box girder increases, the error will also increase [4]. Therefore, shear lag is essential as a part of bridge design, especially when the width of the bridge deck increases.

Since the shear lag effect has been widely recognized in the engineering field, many analysis methods have been proposed, such as the finite longitudinal beam method [5], energy variation method [6, 7], and finite beam element method [8]. Although the accuracy is higher, these methods are too complex to be applied to engineering. Therefore, the concept of effective width is introduced to simplify calculation and analysis. The effective width is the reduced width of the cross-section with a uniform normal stress distribution. It is used to replace the actual width in a simplified analysis based on the elementary beam theory, and the normal stress within the effective flange width must be the maximum normal stress, which likely occurs near the web-flange interaction at the critical sections with peak moments to satisfy strength-design requirements [9]. This concept is recommended by various codes and standards, including AASHTO LRFD Bridge Design Code, British Standards Institution (BSI) BS 5400, and so on. These codes provide some formulas and empirical curves for different support conditions to simplify calculations by considering only the section width \((b)\) and beam span width \((L)\). For example, take the AASHTO LRFD Bridge Design Specifications as representative of these. It provides two empirical curves of effective flange width coefficients, which are equal to the effective width divided by the physical width. The physical width is the ratio of width to the notional girder span \((b/L)\) [10]. Besides the AASHTO LRFD specifications, the Specifications for the Design of Highway Reinforced Concrete and Prestressed Concrete Bridges and Culverts (JTG 3362-2018) and the Steel, Concrete, and Composite Bridges (Part 5) also considered the section width and beam span width alone.
A similar situation is also reflected in some studies. Yuan et al. [11] found that the width of the concrete slab \( (b) \), the span of the beam \( (L) \), and the thickness of the floor \( (h_c) \) are the most relevant parameters and proposed design formulas based on them. Gara et al. [12] proposed several effective width expressions for composite continuous bridges obtained by linear and quadratic polynomial regression. These expressions are functions of beam span \( (L) \), plate width \( (B) \), and beam spacing \( (B_h) \), and each expression contains at least five coefficients. Qin & Liu [13] obtained the effective width expression of closed polynomial under uniform load based on the symplectic elastic method. In the expression, Poisson’s ratio \( (\mu) \) and flange plate width \( (b) \) are considered.

It is easy to find that almost all existing codes and studies only consider the beam span \( (L) \) and plate width \( (B) \). However, the shear-lag effect will be significantly affected by the type of load, boundary conditions, and geometric characteristics (such as web size, flange thickness, etc.). Therefore, the results produced by these formulas may not be conservative [9]. In addition, these formulas and empirical curves are based on the constant depth box girder as the research object. Hence, the direct use of these formulas may be inaccurate for variable depth box girders. Besides, calculating effective width from formulas or curves is still time-consuming. The design efficiency can be improved if the effective width can be predicted rapidly.

In the past few decades, deep learning, a specific machine learning method based on artificial neural networks (ANNs), has made remarkable achievements in many fields, especially in computer vision and language modeling [14, 15]. Many researchers have demonstrated that ANNs can find non-linear mappings between inputs and expected outputs [16]. Therefore, some recent studies use neural networks to learn the relationship between the structural design variables and performance. Nguyen et al. used ANN instead of the finite element method (FEM) to predict the maximum horizontal deformation of the structure under earthquake action [17]. Wu and Kareem used an ANN to predict the acceleration response of the box section under turbulent wind flow [18]. Liang et al. developed a deep learning (DL) model, which was designed and trained to accept FEM input and directly output the aortic wall stress distribution, bypassing the FEM calculation process [19]. All these studies show that compared with FEM, the ANN method has sufficient sharpness and is more advanced in terms of computational time and consumption.

Given the above, the purpose of this article is to predict the effective width of a varying-depth girder bridge through a convolutional neural network. Firstly, establish a parameter dataset to model thousands of finite element models, and provide effective width and effective width coefficients. Then, these results are combined with the parameter dataset to form a dataset that is further divided into three parts: training set, validation set, and test set. The design parameters are converted into two-dimensional data forms to express the geometric shape of the bridge, which can be processed by the convolutional neural network. Subsequently, a convolutional neural network was established to predict the effective distribution width, and an artificial neural network was established for comparison. The low error indicates that the CNN model can predict the effective width with high accuracy. The novelty of this method is that the trained CNN model can predict the effective width in 1 second, allowing for preliminary real-time analysis. Since CNN is a data-driven method, it is considered a “black box” model. Therefore, the CNN model cannot replace FEM for final analysis, however, it can make the design process more efficient. As far as the author knows, this method has not appeared before.

2. Method

2.1. Bridge Parameters Dataset. The basis for accurate prediction through artificial neural networks is a large amount of data on the beam bridges of different depths. The amount of data depends on the size of the ANN model, usually thousands of data. Using real bridge data is an optimistic choice; however, it is difficult to collect such a large amount of data. Another option is to generate a dataset based on real bridge data. In the literature [20], the author evaluated 349 continuous rigid frame bridges and analyzed the design parameters. Based on these data, this article created a dataset.

Before creating the dataset, one thing needs to be done. As shown in Figure 1, a simplified assumption is made that the waist of the box girder is ignored. Furthermore, prestressed tendons and steel bars are also ignored [21, 22]. The design parameters analyzed in [20] include span, girder bottom curve parameters, bridge pier, midspan span-to-height ratio, the thickness of the bottom and top slab, and the width of the top and bottom slab. The width of the roof is slightly larger than the width of the road surface, and continuous rigid frame bridges are mostly built on high-grade highways. Therefore, the width of the top plate is assumed to range from 10 to 20 m. The width of the bottom plate depends on the width of the flange. According to design experience, the flange width is generally above 5 m, which is less than half of the beam width. All parameters used in this article are shown in Table 1.

2.2. Finite Element Method Model. It is a common practice to use the finite element method to analyze the accurate effective width. There are two ways to implement finite models, two-dimensional or three-dimensional [23]. By introducing some assumptions, the web and flange are modeled in two dimensions. However, the fewer assumptions imposed on the analysis, the closer the analysis model is to reality, and a better stress concentration factor will be obtained [24]. Therefore, based on the parameters in the dataset, 2000 finite element models of 3D shell elements with beams of different depths were created. To simplify the model and reduce the computer workload, only the single beam of the main span is modeled, and the bridge piers are ignored like the foundation. Therefore, a fixed boundary is applied at both ends of the beam to simulate the effect of the bridge pier. In addition to geometric parameters and supporting conditions, how to apply the load is also a crucial factor. Different load types, such as concentrated load and
uniform load, lead to different stress distributions and ultimately different effective widths. In this study, the main purpose is to explore a method to predict the effective width using a neural network instead of parameter analysis. Hence, only gravity is applied in the model. For a specific load or load combination, the method proposed in this article can be modified, and a neural network model corresponding to the load can be created. In addition, for the reasons mentioned in Section 2.1, steel bars and prestressed tendons are also ignored.

The finite element models mentioned above are all implemented with ANSYS software, and a script is written in Python to assist in the completion of the work. Shell 63 units are used for the concrete, and the elastic modulus, Poisson’s ratio, and density of the concrete are set to 3.45 × 10^4 Mpa, 0.2, and 2500 kg/m^3, respectively. Mesh generation also plays an important role in finite element modeling. The sparser the grid, the lower the calculation accuracy. On the other hand, a denser grid can improve accuracy while consuming more calculation time. We test 4 unit sizes, 0.25 m, 0.5 m, 1.0 m, and 2.0 m on a sample bridge. The parameters of the sample bridge are listed in Table 1. Table 2 shows the effective width and effective width coefficient on the pier. It can be seen from the table that when the cell size is 250 mm and 500 mm, the difference between the effective width and effective width coefficient is not large. Hence, we set the unit size to 500 mm, considering the spend time and precision.

Figure 2 shows the stress distribution of the girder along the Z-direction or longitudinal direction. It is easy to find that the stress distribution is uneven, the stress on the rib is the largest, and the farther away from the web, the smaller the stress. According to the definition, the effective width can be expressed as follows:

$$ b_e = \frac{t \int_0^h \sigma(x, y) \, dy}{\sigma_{\text{max}}}, $$

(1)

in which $\sigma(x, y)$ is the function of the normal stress along the girder width, $\sigma_{\text{max}}$ means the maximum normal stress in the top slab, $y$ is the position along with the top slab, and $b$ is the width of the top slab. Besides, a coefficient of the effective width ($\eta$) is defined, which is equal to the effective width divided by the width, as follows:

$$ \eta = \frac{b_e}{b}. $$

(2)

Figure 3 plots the effective width coefficient in relation to width and span. In the figure, $X$-axis is the width of the box girder, $Y$-axis is the span, and the color of the dots represents the values of the effective width coefficient. The darker the color, the greater the effective width factor, and the whiter the color, the smaller the effective width factor. This figure

---

**Table 1: Parameters and their value from the bridge parameters dataset.**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Value</th>
<th>Sample bridge</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Span</td>
<td>$L$</td>
<td>$U(100, 300)$</td>
<td>258</td>
<td>[m]</td>
</tr>
<tr>
<td>Girder bottom curve parameter</td>
<td>$n$</td>
<td>$U(1.5, 2.0)$</td>
<td>1.7</td>
<td>[-]</td>
</tr>
<tr>
<td>Span-to-height ratio at piers</td>
<td>$R_p$</td>
<td>$U(1/18, 1/16)$</td>
<td>0.0595</td>
<td>[-]</td>
</tr>
<tr>
<td>Span-to-height ratio at midspan</td>
<td>$R_m$</td>
<td>$U(1/50, 1/40)$</td>
<td>0.0212</td>
<td>[-]</td>
</tr>
<tr>
<td>Girder height in piers</td>
<td>$H_p$</td>
<td>$R_p \times L$</td>
<td>15.365</td>
<td>[m]</td>
</tr>
<tr>
<td>Girder height in midspan</td>
<td>$H_m$</td>
<td>$R_m \times L$</td>
<td>5.487</td>
<td>[m]</td>
</tr>
<tr>
<td>Thickness of the bottom slab</td>
<td>$T_b$</td>
<td>$U(25, 35)$</td>
<td>29</td>
<td>[cm]</td>
</tr>
<tr>
<td>Thickness of the top slab</td>
<td>$T_t$</td>
<td>$U(25, 35)$</td>
<td>31</td>
<td>[cm]</td>
</tr>
<tr>
<td>Thickness of the web</td>
<td>$T_w$</td>
<td>$U(30, 80)$</td>
<td>52.7</td>
<td>[cm]</td>
</tr>
<tr>
<td>Width of top slab</td>
<td>$B_t$</td>
<td>$U(10, 20)$</td>
<td>15.583</td>
<td>[m]</td>
</tr>
<tr>
<td>Width of the bottom slab</td>
<td>$B_b$</td>
<td>$U(B_g - 5, B_g/2)$</td>
<td>7.357</td>
<td>[m]</td>
</tr>
</tbody>
</table>

Note: $U$ denotes a uniform distribution. The value of the girder bottom curve parameter keeps 1 decimal place, and the values of $R_p$ and $R_m$ keep 3 decimal places, whereas the rest of the values keep the integer.

---

**Table 2: Results of the sample bridge with different unit size.**

<table>
<thead>
<tr>
<th>Unit size (mm)</th>
<th>250</th>
<th>500</th>
<th>1000</th>
<th>2000</th>
</tr>
</thead>
<tbody>
<tr>
<td>Effective width (mm)</td>
<td>15352.011</td>
<td>15349.571</td>
<td>15333.567</td>
<td>15322.764</td>
</tr>
<tr>
<td>Effective width factor</td>
<td>0.9851</td>
<td>0.9850</td>
<td>0.9840</td>
<td>0.9833</td>
</tr>
</tbody>
</table>

---

**Figure 1:** Simplified box girder and design parameters.

**Figure 2:** Stress distribution of the girder along the Z-direction or longitudinal direction.

**Figure 3:** Plot of the effective width coefficient in relation to width and span.
shows the correlation between parameters and effective width.

As can be seen from the figure, with the increase of width, the color of points tends to become white, indicating that the shear lag effect is more obvious. It also proves that the calculation results of finite element model are correct.

2.3. Neural Network Model. Although the above 11 design parameters can describe the geometric shape of the main beam well, they cannot be used directly for convolutional neural networks. Convolutional neural networks are usually used to process two-dimensional data, such as images, or one-dimensional time series data. Therefore, it is necessary to express the geometric shape of the box girder in the form of two-dimensional data.

As the shape of the box girder is controlled by the ridgeline, this article selects points on the ridgeline and then arranges the coordinates of the points to form two-dimensional data. First, select 12 ridgelines to represent the entire main beam, and these 12 ridgelines are marked as a=1, respectively. Then, select 10 points evenly on each ridgeline. The 3 coordinates of these 120 points can form a $12 \times 10 \times 3$ tensor for input into the convolutional neural network. The whole process is shown in Figure 4. Before being fed to the convolutional neural network, the tensor needs to be zero-padding to $12 \times 12 \times 3$ to facilitate convolution operations.
2.3.1. Network Architecture Description. As a machine learning algorithm, the convolutional neural network (CNN) plays an important role in its development. Convolutional neural networks are generally composed of convolutional layers, pooling layers, and fully connected layers.

The convolutional layer contains some convolution kernels of size \( k \times k \), also called filters. Each convolution kernel scans the entire input with a specific stride, and at the same time performs dot product with a local area of the input. The sum of all dot products plus the bias is the output of this convolutional layer. Traditionally, a nonlinear activation function is followed immediately after each convolutional layer so that the network has a nonlinear expression ability.

When processing a large amount of input data, the pooling layer is usually used to reduce the spatial size of the feature map. Popular pooling layers include maximum pooling and average pooling, which obtain the maximum or average value from the pooling window.

In the CNN structure, one or more fully connected layers are connected after passing through multiple convolutional layers and pooling layers. Like multilayer perceptions, each neuron in the fully connected layer is fully connected to all neurons in the previous layer. The full connection layer can integrate features extracted from the convolution layer or pooling layer.

Figure 5(a) shows the CNN architecture used in this article. It is mainly composed of nine convolutional layers stacked. The convolution kernel of all the convolution layers is \( 3 \times 3 \), and the difference lies in their output channels. The output channels of the first 2 layers are 32, the middle 4 layers are 64, and the last 3 layers are 128. For every 4 convolution layers, a pooling layer is inserted to reduce dimension. After the last convolution layer, a full connection layer of 128 neurons is used to output a value. This value can be an effective width or an effective width factor, depending on the training data. Since there is only one output in this model architecture, two models are needed for effective width (or effective width coefficient) on pier and midspan, respectively.

For comparison, a traditional artificial neural network was constructed at the same time. It is a typical multiple layer ANN, including one input layer, five hidden layers with 30 neurons in each hidden layer, and one output layer in total. This model takes all the design parameters as input and predicts one value. Similarly, this value can be the effective width or the effective distribution coefficient.

2.3.2. Data Preprocessing. Once the model architecture is set up, the next step is to train the model. From the data set of bridge parameters and the simulation of the finite model, an input space that includes 9 bridge design parameters and an output space, including effective width \((b_e)\) or effective width coefficient \((\lambda)\), can be obtained. Before passing these data into the model, the data needs to be preprocessed to speed up model training and improve accuracy. Standardization is one of the wildly adopted preprocess methods, which rescales the input space into distribution with \( \mu = 1 \) and \( \sigma = 0 \). All the inputs are calculated as follows:

\[
    z = \frac{x - \mu}{\sigma},
\]

where \( \mu \) is the mean of the samples and \( \sigma \) is the standard deviation from the mean. By doing this, all inputs are centered around 0 with a standard deviation of 1, and the bias because of the large magnitude of some variables can be avoided.

The next step is to divide the dataset into a training set, a test set, and a validation set. The training set is used to fit the model. Validation can be used to adjust the hyperparameters of the model and make a preliminary assessment of the model’s capabilities. The test set is used to evaluate the generalizability of the final model. Firstly, divide the data set into a training set and a test set at a ratio of 8:2. Then, 20\% of the training data is sampled as a validation set to further split the training data. Therefore, the final training set, validation set, and test set have 1200, 400, and 400 samples, respectively, all of which are randomly divided.

2.3.3. Training of the Models. The parameters of the neural network model are learned from the training data. The network training process is a data-sensitive process, which means that different data sets will lead to different prediction accuracy. To this end, all models are trained 5 times to reduce errors, and each model’s training, verification, and test data sets are randomly generated each time, thereby reducing errors caused by datasets and making model evaluation more objective.

During the training process, two performance metrics are used to evaluate the prediction accuracy: mean absolute error (MAE) and mean absolute percentage error (MAPE). For each model, the MAE and the MAPE are defined, respectively, as follows:

\[
    \text{MAE} = \frac{1}{n} \sum_{i=1}^{n} |y_i - y'_i|,
\]

\[
    \text{MAPE} = \frac{1}{n} \sum_{i=1}^{n} \frac{|y_i - y'_i|}{y_i},
\]

where \( y_i \) is the effective width or effective width coefficient from the finite element models, while \( y'_i \) is the prediction from the network.

3. Result and Discussion

The model proposed in Section 2.3 was independently trained 5 times on a dataset of 1600 bridge parameters, effective widths, and effective width coefficients. All learning curves of these 2 models have the same trend but different start values. The learning curve of the CNN model for the first time is plotted in Figure 6. Firstly, the curve starts at a very high value and drops sharply in the first 2 or 3 periods. Since then, the rate of decline begins to slow down. After a slow decline for about 25 epochs, the index remained stable at about 0 and remained until the end of the training.
Figure 4: Process of forming input tensor for CNN. (a) Box grid. (b) Selecting grid. (c) Form the input tensor. (d) Zero-padding to the shape of $12 \times 12 \times 3$.

Figure 5: The model architecture of CNN and ANN. (a) CNN model (b) ANN model.
A lot of information can be obtained from the curve. Firstly, the dramatic changes in the previous periods indicate that the network is effectively learning from the training dataset. On the contrary, the curve changes slightly after the 25<sup>th</sup> epoch, which indicates that the network has reached its best state after 25 epochs and has been fully trained. In addition, the stable trend proves that these models are not overfitting because the curve rebound at the end of the training is a common phenomenon of overfitting the network. In addition, the same trend of the training set and validation set curves also shows that these models have good generalization ability, which means that these models can also work well on strange data sets.

Similarly, these models are independently tested 5 times in the test dataset after training. Since the effective width and the effective width coefficient have different sizes and dimensions, it is impossible to directly discuss their accuracy by comparing the difference between the predicted value and the test value. Hence, the percentage error was chosen, and for effective width and effective width coefficient, the percentage error is defined as follows:

\[
\delta_b = \frac{b_{\text{pred}} - b_{\text{test}}}{b_{\text{test}}} \times 100\%
\]

\[
\delta_\eta = \frac{\eta_{\text{pred}} \times b - \eta_{\text{test}} \times b}{\eta_{\text{test}} \times b} \times 100\%
\]

(5)

Then, some statistical parameters of the percentage error of the two models, including the mean percentage error, the maximum percentage error, and the variance of the percentage error, were calculated and listed in Table 3.

Table 3 shows the average percentage error of the 2 models at different positions and different outputs. It is not
difficult to find that the errors of all these 2 models are relatively small. The maximum value was only 0.619 of the ANN model on the pier, with effective width as output. In other words, the accuracy of all these models is very high, about 99%. In particular, the average percentage of models with effective width coefficients as output is less than 0.5%. At the same time, when the model uses the effective width coefficient instead of the effective width as the output, all average percentage errors will be reduced, although the decline is different for different models. The reason for the reduction is that the effective width factor is a series of numbers between 0 and 1. Therefore, replacing the effective width \( \eta \) with the effective width coefficient \( \eta \) reduces the error caused by the different magnitudes of effective width, like the standardization in the preprocessing. In addition, the greater the error, the greater the reduction.

The ANN model with an effective width as output has the highest maximum percentage errors, which are 6.263. Except for this, the maximum error of the other models is about 5%, which is acceptable in the engineering field. All models with effective width coefficients as output have an error of less than 2%.

For the variance of the percentage error, all variances are less than 1, indicating that the error distribution is centered on the average percentage error and that the model has stable prediction accuracy. The same rule as the average percentage error can also be found in Table 3. When the effective width coefficient is used instead of the effective width, the maximum percentage error and the percentage variance are reduced.

Comparing the average percentage error of the 2 models with the effective width coefficient as the output, it can be found that the percentage error of the CNN models is lower. The reduced error between the ANN model and CNN model shows that convolutional layers can effectively improve prediction accuracy.

In addition to the above, the performance of the models to predict effective width coefficients at the pier section on different size datasets was tested. The amounts of data are 500, 1000, 1500, 2000, respectively.

The Table 4 shows the mean absolute percentage errors of the two models. As can be seen from the table, when the amount of data is greater than 1000, ANN’s prediction error tends to be stable while CNN’s error is still decreasing. It is because CNN has more trainable parameters. Hence, it has stronger nonlinear fitting ability. When the amount of data increases, the CNN model can have better prediction performance.

All these models are implemented using TensorFlow. Given the design parameters, the model can output the effective width factor within 1s. For comparison, FEA takes about 30 minutes to obtain the effective width factor for the same input. Although training a model is a time-consuming process, it is much faster to use the model to predict the effective width. However, once the model learns the relationship between the output and input of the FEM, the training process is no longer needed. Compared with the existing effective width formula, the model considers more factors, thus predicting more accurate results.

As a feasibility study, the bridge design parameters are based on the investigation of the continuous reinforced concrete rigid frame bridge in China and some design experience. Therefore, inputs that are significantly different from the training data may cause large errors in the model. According to research on the relationship between deep learning models and training data, this problem can be solved by a larger training dataset [25]. In addition, some factors, such as prestress and re-bar, were not considered in the finite element modeling. This problem can be solved by more refined models or test data on actual bridges.

This method will have a huge impact on bridges design. Because of the similar geometric shapes of the box girder bridges of varying depths, the range of design parameters is limited. Hence, this method can be applied in most cases. For other types of bridges, a similar model can be created and trained like this method to predict the effective width. More importantly, this method will enable real-time effective width analysis, thereby accelerating safe and economical bridge design. In addition, this method can be combined with some optimization algorithms to optimize the bridge design. Designers no longer have to spend a lot of time trying. After the preliminary design is determined, the finite element model is used for accurate calculation. Otherwise, it can be combined with some BIM software to provide real-time effective width display so that the designer has an intuitive understanding.

### 4. Conclusions

In this article, a method of using the advantages of artificial neural networks to predict the effective width of box girder bridges with different depths is proposed. A multilayer feed-forward artificial neural network model and a CNN model are created. Establish the bridge design parameter dataset and finite element dataset, and train, verify, and test the model. These
models use bridge design parameters as input to predict the effective width and generally show good agreement. The lower errors in the test set indicate that the CNN model can be used to predict the effective width. By comparing the training results on the datasets of different sizes, it can be found that when the amount of training data is small, ANN is better, and when the amount of data is large, CNN is better. Furthermore, by comparing the prediction errors of the various models, it can be found that using the effective width coefficient as output instead of the effective width will effectively improve the accuracy.

Compared with the finite element model, the proposed CNN model provides more efficient calculations with the same accuracy. This method will be able to analyze the effective width in real-time and combine it with other technologies, such as optimization algorithms to play a greater role.

This paper demonstrates the advantages of CNN in predicting effective width. However, the proposed model requires sufficient data for proper CNN training before it can be used for prediction. When the input data exceeds the training set data range, it will be unreliable. In addition, the accuracy of the finite element model is also an important factor, however, this article has made some simplifications.

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.

References