Multiscale Science and Engineering (2023) 5:53–61
https://doi.org/10.1007/s42493-023-00094-3

ORIGINAL RESEARCH

Predict Elastic Properties of Fiber Composites by an Artificial Neural Network

Hao-Syuan Chang[1] · Jia-Lin Tsai[1]

Received: 17 April 2023 / Revised: 29 May 2023 / Accepted: 5 June 2023 / Published online: 24 June 2023

Abstract

An artificial neural network was utilized to predict the elastic properties of fiber composites with varying fiber positions and volume fractions. Randomly distributed fibers were placed in repeating unit cells (RUC) to represent the microstructures of the composites, with fiber volume fractions ranging from 41 to 60%. The effective elastic constants of the RUC were determined using finite element analysis (FEA). To process the binary images of the microstructure, a convolutional neural network (CNN) model was employed in the study. A total of 4320 datasets were created, with 90% being used for training and the remaining for validation. Additionally, the trained CNN model was used to process microstructural images obtained from experiments and literature. Results from the artificial neural network were compared to those from FEA, with an average difference of around 5%. To further reduce this discrepancy, transfer learning was applied to the trained CNN model. After transfer learning, the difference between the label value and the CNN prediction decreased to 2%. Therefore, the artificial neural network model proved to be an effective method for characterizing the elastic constants of fiber composites.

Keywords Fiber composites · Elastic constants · Transfer learning · Convolutional neural network

Introduction

Long fiber composites are widely used in industry, but it takes a lot of time and cost to obtain the mechanical properties through experiments and simulation. Therefore, building a neural network model has become a new trend to predict the mechanical properties of fiber composites using artificial intelligence. For example, the artificial neural network (ANN) [1] is a computing model that mimics the neurons of the human brain. It can fit the neural network model parameters through input numerical information to establish nonlinear function approximation of input and output and achieve optimal estimation. Kim et al. [2] generated long fiber models with fixed fiber quantities using the random sequential expansion (RSE) method [3], and produced the models with different fiber volume fraction by changing the fiber radius. They solved the equivalent modulus of the composites through ABAQUS commercial software with periodic boundary conditions [4, 5]. Subsequently, they used the fiber centroid co-ordinates and volume percentage as input and the equivalent modulus as output to establish the ANN neural network. They also compared the effect of database size on prediction accuracy. Results showed that the prediction error was mostly less than 1%, and the correlation coefficient was above 0.96 [6]. Moreover, the accuracy of the model increased with larger database training. Although the ANN neural network model can accurately predict the equivalent modulus of composite materials, when the fiber quantity in the fiber model changes, the input values and neuron quantities may not match, leading to model failure. The convolutional neural network (CNN) [7, 8] can analyze input images without this limitation. Therefore, Chen et al. [9] used glass and graphite fibers as reinforcing materials in composite materials, and used a two-dimensional convolutional neural network to analyze the equivalent modulus of randomly distributed long fiber composite materials. They predicted the Young's modulus, shear modulus, Poisson's ratio, and thermal expansion coefficient in three directions by analyzing the fiber distribution position in the cross-sectional shape. The results showed that regardless of the fiber type, the median absolute prediction error was about 2%. In addition to predicting the equivalent modulus of composite materials, convolutional neural networks are also suitable

✉ Jia-Lin Tsai
jialin@nycu.edu.tw

[1] Department of Mechanical Engineering, National Yang Ming Chiao Tung University, Hsinchu 300, Taiwan

for other engineering applications. Kim et al. [10] generated unidirectional fiber models with the consideration of fiber matrix interfacial debonding and simulated the stress–strain curve of transverse tensile test using finite element method. They divided the stress–strain curve equally into 40 parts based on strain values and used corresponding stress values as target values. They input the fiber cross-sectional shape into the convolutional neural network (CNN) to predict the stress at each strain. Sorini et al. [11] developed a long fiber model with variable fiber and matrix material properties. They used the high-fidelity generalized method of cells (HFGMC) [12] micromechanical analysis of the composite material's equivalent stiffness matrix as the target value, while with fiber cross-sectional images as input and the convolutional neural network was used to predict the equivalent stiffness matrix values. The results showed that most predicted values were close to the label values, but the analysis time was 25,000 times faster, indicating that neural networks can significantly reduce the time cost of numerical analysis.

Although neural network models can quickly and accurately obtain the mechanical properties of materials, generating sufficient training data often requires a significant time cost. Therefore, transfer learning methods, which involve training a pre-trained neural network model with additional databases, were developed to improve the accuracy of neural networks. Shin et al. [13] used image recognition convolutional neural networks as an example to explore how to enhance the accuracy of neural networks through different methods, including transfer learning. By fine-tuning the parameters in the neural network through transfer learning, and comparing the prediction errors of different neural network models, it was found that the transfer learning neural network model had better prediction performance than the original model, demonstrating that transfer learning can enhance the prediction ability of neural network models. Jung et al. [14] calculated the stress–strain curves of particle and short fiber composite materials under axial tension and cyclic loading, and used material properties and stress–strain curves as inputs and outputs for training deep neural network models. They also built a transfer learning database through the finite element method, and fine-tuned the parameters of the pre-trained neural network model with a small amount of data. The results showed that the deep neural network model fine-tuned through the transfer learning database had a coefficient of determination [15] increased from 0.9744 to 0.9966, indicating that the transfer learning model had better prediction ability.

In this study, the CNN model was trained by the dataset generated from the finite element method and then modified by the transfer learning dataset. The microstructural images directly obtained from experiment and literatures were used to validate the trained CNN model. The accuracy of the CNN model after the transfer learning process was discussed.

CNN Model

Generate Dataset for CNN Model

In order to generate a database for the CNN model, we created the fiber composites master models at first. The master models were generated by setting a 100-unit square model and randomly filling it with fibers with a diameter of 7.5 units until no new fibers could be placed within the model range. In addition, contact between fibers or with the frame was avoided in the master model. Three sets of main models are shown in Fig. 1, each with a side length of 100 units and containing 137, 133, and 127 fibers with different distributions, with volume percentages of 60.5%, 58.7%, and 56.5%, respectively. To increase model diversity, each set of main models was mirrored and flipped in three directions, as shown in Fig. 2. Taking the 90° mirror image as an example, the x-co-ordinate of the fiber center is obtained through a formula to obtain the new fiber center x-co-ordinate, while the y-co-ordinate remains unchanged, forming the new center co-ordinates, which is the 90° mirror image fiber model. The 0° and 45° mirror images are obtained in the same way to allow fibers to appear at different positions within the model range and to avoid the time and cost required to re-design the main models. Random fiber removal was then performed using 10 different random

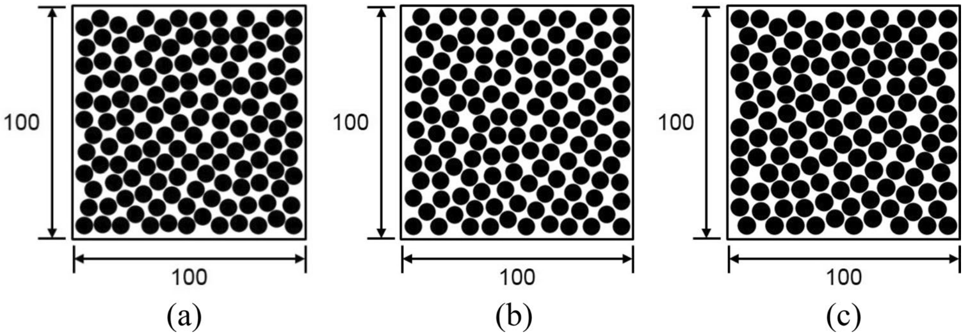


Fig. 1 Three master microstructures **a** $V_f=60.5\%$, **b** $V_f=58.7\%$, **c** $V_f=56.5$

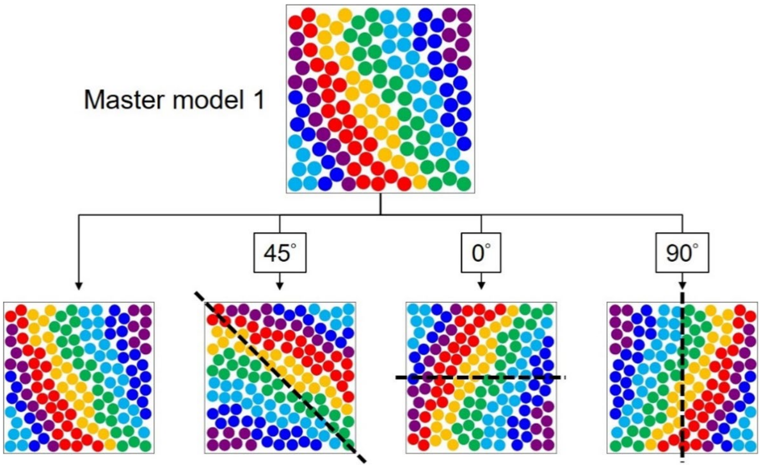


Fig. 2 Four sub-microstructures were generated from the master microstructure through mirror image process

seeds, by assigning numbers to the fibers in the model and removing them in different order. The matrix was utilized to fill the gaps created by the removed fibers. A total of 35 fibers were removed, resulting in 36 different models with different fiber volume fractions and distributions, as shown in Fig. 3. Through this method, a total of 4320 microstructural models of fiber composites was generated for the database, with fiber volume percentages ranging from 41 to 60.5%.

Subsequently, a fiber structure matrix was generated through image pre-processing to present the cross-sectional structure of the fiber model and serve as input for the neural network model. The fiber cross-section image with extension of image JPG was imported into Python 3.8.8. At this time, the image is a three-color image (RGB figure) with a size of 560×560 pixels, as shown in Fig. 4a. The color of each pixel cell in the image was represented by three colors: red, green, and blue, and the color intensity is represented by values ranging from 0 to 255. Afterwards, the image was converted to a grayscale image using grayscale conversion [16], as shown in Eq. (1):

$$I_y = 0.333F_r + 0.5F_g + 0.1666F_b \tag{1}$$

where, I_y represents the grayscale value, F_r, F_g, and F_b represent the intensity of red, green, and blue, respectively. At this time, the pixel cells in the image were presented by grayscale

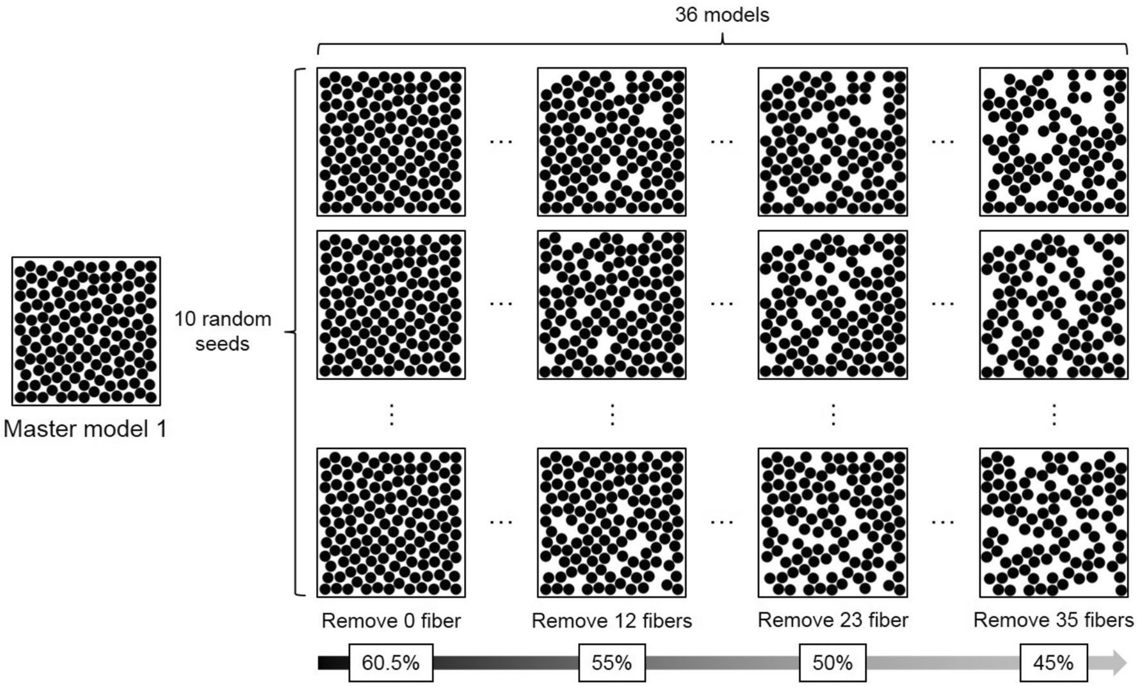
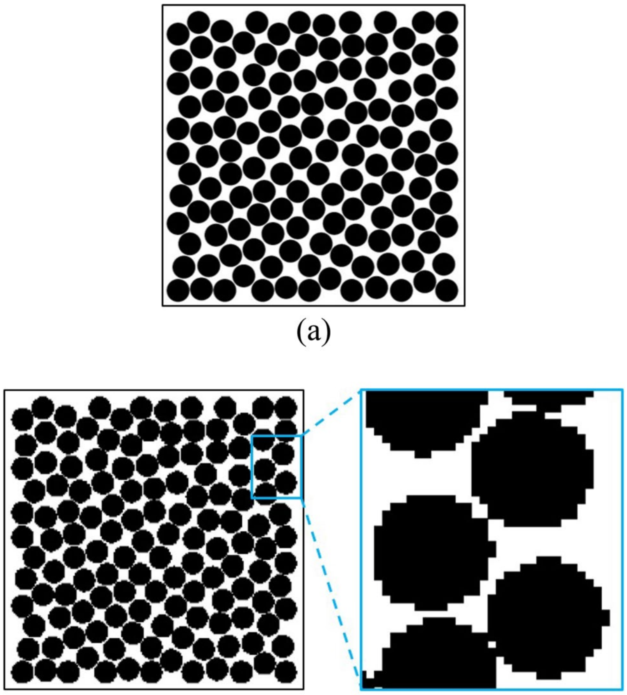


Fig. 3 The fibers were removed from the sub-microstructure gradually based on random seed process


(a)

(b)

Fig. 4 The image of the microstructure **a** before binarization, **b** after binarization

values ranging from 0 to 255. The matrix material is white with a grayscale value of 255, while the fiber is black with a grayscale value of 0. The pixel cells at the interface between the fiber and matrix will appear as gray with varying depths depending on the ratio of the fiber and matrix. The grayscale values range from 1 to 254 to define the boundary position of the fiber. The binary images of the microstructures were created through the binarization thresholding process [17] with the proper threshold value. By changing the proper threshold value, the proportion of fiber in the binary image output can be made to approach the volume percentage of the fiber composite model. The threshold value was set to 200 as the boundary standard and the result is shown in Fig. 4b. Then, the image size was unified to a matrix of 200 × 200 through the nearest interpolation [18] to fit the neural network input size. Subsequently, the Min–max normalization [19] was used to normalize the fiber pixel cells to 0 and the matrix pixel cells to 1 to avoid a large difference between the matrix values and the equivalent modulus values, which can make it difficult for the neural network to converge, while reducing the required resources and training time. Finally, the fiber structure was stored as a matrix form for the database.

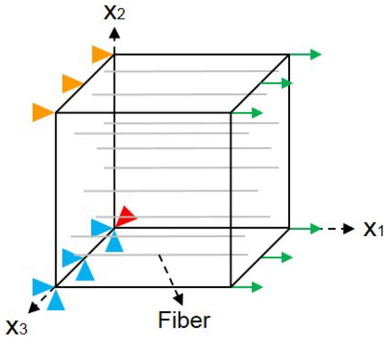


Fig. 5 Boundary condition for the calculation of Young's moduli in FEM analysis

Finite Element Analysis

The equivalent properties for the aforementioned microstructures were calculated through finite element analysis. The fiber centroid co-ordinates were entered into the ANSYS (2016 version) numerical software to construct a finite element RUC model. By utilizing periodic boundary conditions, the three corresponding periodic surfaces of the model exhibited the same local strain field. Assuming that the RUC model has n nodes on the periodic surface with the normal vector x_1, the relative displacement of all nodes on surfaces + 1 and − 1 can be expressed as [4, 5]

$$^{(+1)}u_i^{(n)} - {}^{(+1)}u_i^{(1)} = {}^{(-1)}u_i^{(n)} - {}^{(-1)}u_i^{(1)} \tag{2}$$

where $^{(+1)}u_i^{(n)}$ represents the global displacement vector of node n on the periodic surface + 1, while $^{(-1)}u_i^{(n)}$ represents the global displacement vector of node n on the periodic surface − 1. Similarly, the same formula can be derived for the other two sets of periodic surfaces. The FEM model assumes isotropic materials with a fiber Young's modulus of 240 GPa and a Poisson's ratio of 0.3, and a matrix Young's modulus of 4 GPa and a Poisson's ratio of 0.3. The model uses SOLID187 elements, which are 10-node tetrahedral elements with degrees of freedom in the x, y, and z directions. A finite element model with dimensions of 2 × 100 × 100 is constructed, and a cylinder with a diameter of 7.5 units and a length of 2 units is generated at the fiber centroid co-ordinates to represent the fiber within the composites model.

To calculate the equivalent Young's modulus E_1, we considered a cube model with edge length a, where the displacement is applied in the x_1 direction. The loading is applied as shown in Fig. 5 where the line segment that intersects the $x_1 = 0$ plane and the $x_2 = a$ plane was constrained in the x_1 direction. The line segment that intersects the $x_1 = 0$ plane and the $x_2 = 0$ plane was constrained in both the x_1 and x_2 directions, and the origin was constrained in the x_1, x_2, and x_3 directions. Two line segments intersecting the $x_1 = a$ plane and the $x_2 = 0$ plane, as well as the $x_1 = a$ plane and the $x_2 = a$ plane, were given displacement in the x_1 direction. The formula for the equivalent Young's modulus is evaluated as follows:

$$E = \frac{\sum V_{element} \times \sigma_{element}}{\sum V_{element} \times \varepsilon_{element}} = \frac{\sigma_{model}}{\varepsilon_{model}} \tag{3}$$

where $V_{element}$ represents the element volume, $\sigma_{element}$ represents the normal stress on the element, $\varepsilon_{element}$ represents the normal strain on the element, σ_{model} represents the normal stress on the model, and ε_{model} represents the normal strain on the model. The similar procedure was employed for the equivalent moduli E_2 and E_3. For the equivalent shear modulus G_{12}, we take the $x_2 = a$ as the shear plane. The loading is set up as shown in Fig. 6, where the x_1 direction on the intersection line of $x_1 = 0$ and $x_2 = a$ is constrained, while both x_1 and x_2 directions are constrained on the intersection line of $x_1 = 0$ and $x_2 = 0$. The origin ($x_1 = x_2 = x_3 = 0$) is constrained in all three directions. Two lines on the intersection of $x_1 = a$ and $x_2 = 0$ and $x_1 = a$ and $x_2 = a$ are given displacement in the x_2 direction. The equation for the equivalent shear modulus G is shown below:

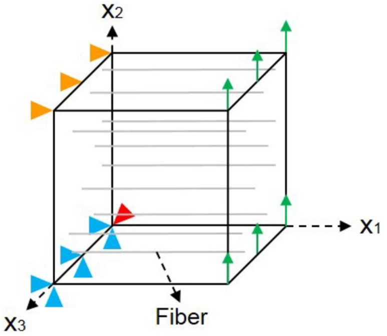


Fig. 6 Boundary condition for the calculation of shear moduli in FEM analysis

$$G = \frac{\sum V_{element} \times \tau_{element}}{\sum V_{element} \times \gamma_{element}} = \frac{\tau_{model}}{\gamma_{model}} \tag{4}$$

where $\tau_{element}$ is the shear stress of the element, $\gamma_{element}$ is the shear strain of the element, τ_{model} is the shear stress of the model, and γ_{model} is the shear strain of the model. We applied the similar procedure to calculate the equivalent shear moduli G_{13} and G_{23}. The obtained values of the equivalent modulus were employed as the output dataset for training the neural network.

Training and Validation of CNN model

When the database design was completed, we began to construct the neural network model. During the training phase, 90% of the database, a total of 3888 sets, were used as training data. The neural network model CNN has a structure as shown in Fig. 7, consisting of three convolutional layers, each with 32 3 × 3 filters with a stride of 1. Padding was set to maintain the same size of input images and output feature maps, and rectified linear units (ReLU) [20, 21] were used as activation functions for the model. A pooling layer (MaxPooling) [22] was added after the first and second convolutional layers to reduce the dimensionality of the output feature maps. A filter size of 2 × 2 was used, and the stride was set to 2. The output data from the third convolutional layer was flattened to form a one-dimensional feature map, which was then inputted into the fully connected layer. The first fully connected layer had 50 neurons to receive the feature vector outputted from the front end, and the second layer had 6 neurons corresponding to the 6 equivalent modulus values. The dropout rate [23] was set to 0. The mean absolute error (MAE) [24] was used as the loss function, and the optimizer used was the Adam gradient descent method [25] with a learning rate decay [26]. The initial learning rate was set to 0.001, and if the training error did not decrease after three epochs, the learning rate was reduced by half. The minimum learning rate was set to 10^{-5}. Finally, hyper-parameters for the neural network were optimized, with a batch size of 30.

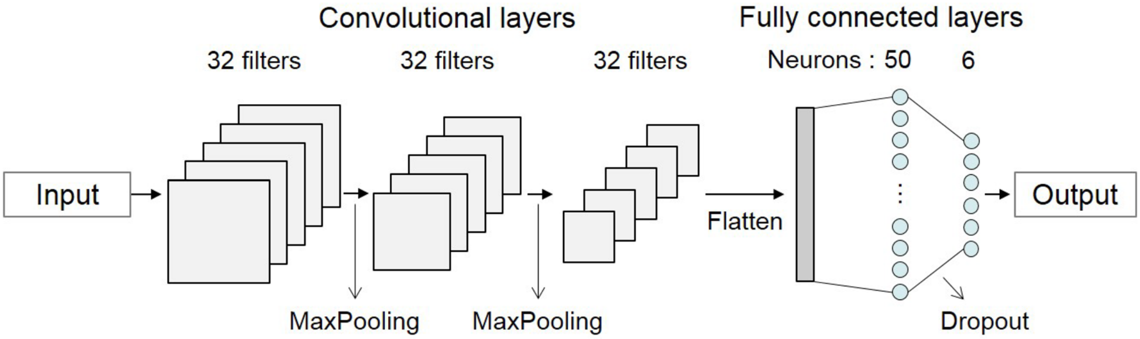
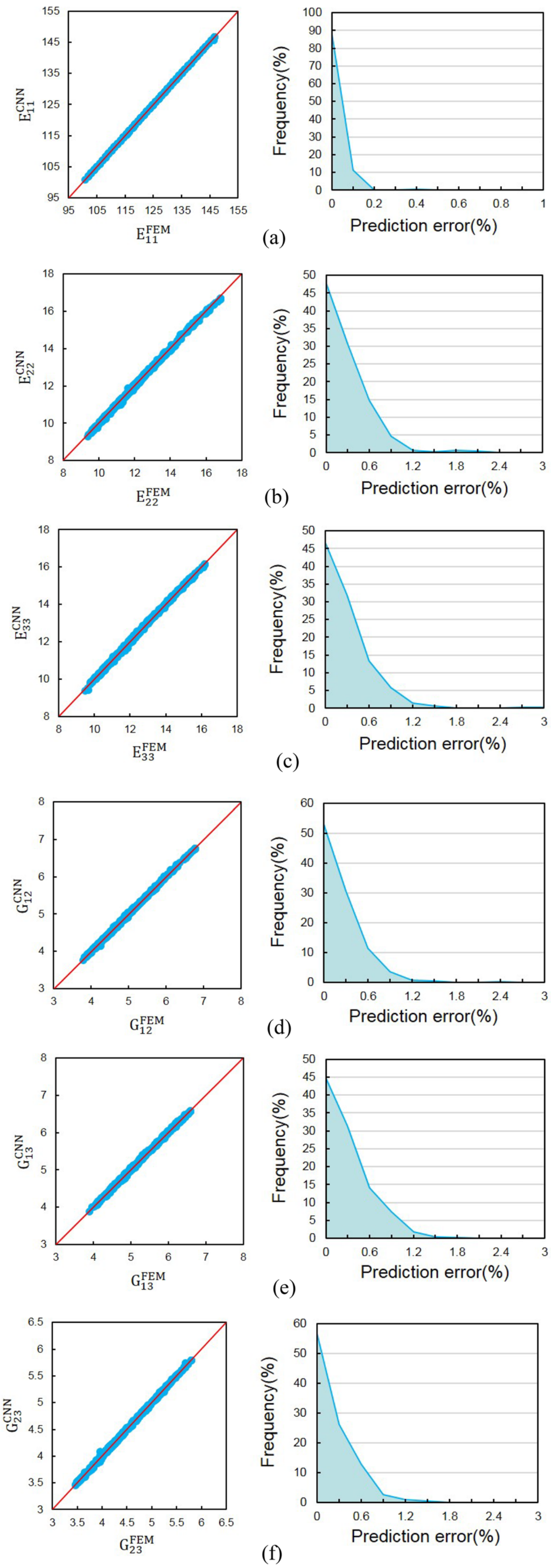


Fig. 7 The structure of CNN model

Fig. 8 Validation of CNN model and prediction error frequency for **a** E_{11}, **b** E_{22}, **c** E_{33}, **d** G_{12}, **e** G_{13}, **f** G_{23}



The data was shuffled and regrouped when entering the next epoch, and a total of 80 training epochs were used. The training was conducted on a computer with an Intel(R) Core(TM) i7-6700 CPU @ 3.40 GHz, and the training time was 93 min.

During the validation phase, 10% of the validation data is fed into the trained neural network model, and the difference between the label values and predicted values were calculated using the mean absolute percentage error (MAPE) [27]. The comparison of label and predicted values and the error frequency are shown in Fig. 8a–c. It can be seen that the average error is within 0.4%. For shear moduli, the label and predicted values are shown in Fig. 8d–f and the average error is around 0.42%. Thus, the results validate the applicability of the trained CNN model for the predictions of the fiber composites.

Testing the Microstructures from Experiments

Subsequently, two sets of cross-sectional images of long-fiber composite materials were obtained through journals and optical microscopy (OM) as test data. The two sets are Microstructure-1 [28] and Microstructure-2, as shown in Fig. 9 respectively. The label values of these images were calculated through finite element method. In addition, the fiber structure matrices were generated through image preprocessing and were then used as input to the convolutional neural network for prediction. The accuracy of the neural network predictions was then compared with the label values to explore its predictive ability. Table 1 shows the MAPE error between the label values and predicted values for the selected microstructures. The average prediction error for Microstructure-1 was 3.53%, while that for Microstructure-2 was 3.32%. The results indicate that the convolutional neural network model can still maintain a certain degree of predictive ability.

In order to enhance the accuracy of the model prediction, we adopted the transfer learning technique in the CNN model. Transfer learning is a method of retraining a pre-trained neural network model using a small dataset. This approach involves fine-tuning the parameters within the pre-trained neural network to increase its generalizability, reduce prediction errors, address difficulties in collecting large datasets, and shorten the overall training time required for the network.

In this study, a transfer learning database was created by generating random distribution long fiber RUC models using Material Designer. Material parameters were set to be the same as before, and a single-direction long fiber

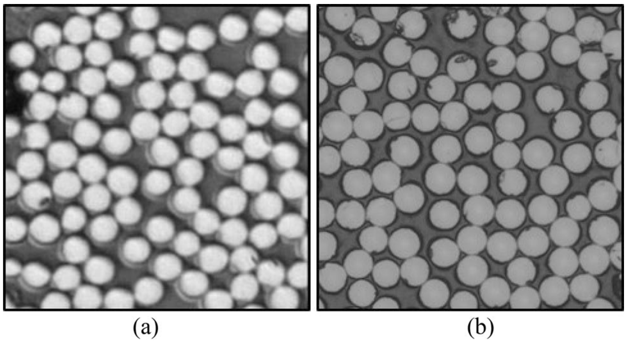


Fig. 9 Microstructural images obtained from **a** literature [25] (microstructure-1), **b** experiment (microstructure-2)

model was used as the reference, with a fiber diameter of 7.5 units, an inclination angle of 0°, a model cross-section of 100 × 100 units, and a default thickness value. When the volume percentage exceeded 40%, the model was prone to meshing failures due to fiber proximity, so models with 64 and 81 fibers were designed, with volume percentages of 28.24% and 35.74%, respectively. fifty different models were generated for each. Tetrahedral 10-node elements with x, y, and z degrees of freedom were used, and periodic boundary conditions were applied to obtain label numerical values. The fiber structure matrix was then generated using image preprocessing. The Material Designer output model image was first cropped using Python to produce a 412 × 412 pixel RGB image. It was then converted to a gray level image with values ranging from 0 to 255. The gray value of the base material in the image was 213. All pixel values with a gray value of 213 were set to 0, while the remaining pixel values were set to 1, marking the positions of the base material and fibers in the model to form a fiber structure matrix. A transfer learning database consisting of 100 sets of data was then created.

The hyper-parameters and parameters in the trained binary neural network model were loaded into Python as the source model. The training hyper-parameter settings were kept the same as the original model, except for batch size and epoch. The transfer learning database was then inputted into the source model for transfer training, with a batch size of 25 and 150 epochs. Fine-tuning was performed on the weights and biases of the neural network model's fully connected layer using the transfer learning database, allowing the transfer learning model to learn the structural characteristics of the fiber models in the transfer learning database. The training time using a computer with the same configuration was approximately 8 min.

Results and Discussion

By using the two microstructure models mentioned above, a transfer learning model (CNN-T) was tested to explore its predictive accuracy. Table 2 shows the MAPE error between the label values and predicted values for both models. The average test error for microstructure-1 was 3.25%, while the average prediction error for microstructure-2 was 2.44%. These results indicate that the transfer learning model has lower prediction errors than the original model for both microstructure models, demonstrating that transfer learning can effectively enhance the predictive accuracy of neural network models for microstructure models and can be applied in practical engineering applications. It is noted that the equivalent moduli of fiber composites could be estimated by simple area proportion method, so called rule of mixture. The method only considers the volume fraction of the fibers instead of the distribution of the fibers. We

Table 1 Comparison of CNN model to the label values for unidirectional composites with two different microstructures

CNN		Elastic constants (GPa)						AVG error (%)
		E11	E22	E33	G12	G13	G23	
Microstructure-1	Label	114.81	11.5	11.64	4.63	4.72	4.32	
	CNN	112.99	11.15	10.63	4.49	4.76	4.16	3.53
Microstructure-2	Label	112.18	11.2	11.05	4.51	4.44	3.95	
	CNN	111.74	10.72	10.27	4.44	4.56	4.1	3.32

Table 2 Comparison of transfer learning CNN model (CNN-T) to the label values for unidirectional composites with two different microstructures

CNN-T		Elastic constants (GPa)						AVG error (%)
		E11	E22	E33	G12	G13	G23	
Microstructure-1	Label	114.81	11.5	11.64	4.63	4.72	4.32	–
	CNN-T	117.63	12.26	12.33	4.61	4.77	4.45	3.25
Microstructure-2	Label	112.18	11.2	11.05	4.51	4.44	3.95	–
	CNN-T	108.97	10.87	11.23	4.49	4.51	4.16	2.44

calculated E_{11}, E_{22} and G_{12} for the microstructures 1 and 2, respectively. For the microstructure 1, the calculated values are $E_{11} = 114.81$ GPa, $E_{22} = 8.16$ GPa and $G_{12} = 2.86$ GPa; for the microstructure 2, the corresponding values are $E_{11} = 112.18$ GPa, $E_{22} = 7.97$ GPa and $G_{12} = 2.5$ GPa. As compared to the data shown in Table 2, it can be seen that the results obtained from the rule of mixture, except the value of E_{11}, deviate from FEM (label) solutions and the CNN predictions. Thus, the neural network model illustrates the advantage for characterizing the mechanical properties of the fiber composites with accuracy, especially in the transverse and shearing directions.

Conclusion

The average prediction errors of the convolutional neural network (CNN) for the two microstructure models were 3.53% and 3.32%, respectively, while those of the transfer learning model (CNN-T) were 3.25% and 2.44%. It can be observed that the transfer learning model had lower prediction errors than the CNN, indicating that it can effectively learn the relationship between the fiber model features and the equivalent modulus in the new database. This enables the transfer learning model to make more accurate predictions for the microstructure models, making it suitable for engineering applications.

Data availability The data that support the findings of this study are available from the corresponding author upon reasonable request.

Declarations

Conflict of interest The authors declare that they have no known competing financial interests that are directly or indirectly related to the work submitted for publication.

References

1. O.I. Abiodun, A. Jantan, A.E. Omolara et al., State-of-the-art in artificial neural network applications: a survey. Heliyon. **4**(11), e00938 (2018)
2. D.-W. Kim, S.-M. Park, J.H. Lim, Prediction of the transverse elastic modulus of the unidirectional composites by an artificial neural network with fiber positions and volume fraction. Funct. Compos. Struct. **3**(2), 025003 (2021)
3. S.-M. Park, J.H. Lim, M.R. Seong et al., Efficient generator of random fiber distribution with diverse volume fractions by random fiber removal. Compos. B. Eng. **167**, 302–316 (2019)
4. Z. Xia, Y. Zhang, F. Ellyin, A unified periodical boundary conditions for representative volume elements of composites and applications. Int. J. Solids Struct. **40**(8), 1907–1921 (2003)
5. Z. Xia, C. Zhou, Q. Yong et al., On selection of repeated unit cell model and application of unified periodic boundary conditions in micro-mechanical analysis of composites. Int. J. Solids Struct. **43**(2), 266–278 (2006)
6. R. Taylor, Interpretation of the correlation coefficient: a basic review. J. Diagn. Med. Sonogr. **6**(1), 35–39 (1990)
7. S. Albawi, T.A. Mohammed, S. Al-Zawi, Understanding of a convolutional neural network. in *2017 International Conference on Engineering and Technology (ICET)* (2017), pp. 1–6
8. N. Kalchbrenner, E. Grefenstette, P. Blunsom, A convolutional neural network for modelling sentences. arXiv preprint arXiv:1404.2188, (2014)
9. Q. Chen, W. Tu, M. Ma, Deep learning in heterogeneous materials: targeting the thermo-mechanical response of unidirectional composites. J. Appl. Phys. **127**(17), 175101 (2020)
10. D.-W. Kim, J.H. Lim, S. Lee, Prediction and validation of the transverse mechanical behavior of unidirectional composites considering interfacial debonding through convolutional neural networks. Compos. B. Eng. **225**, 109314 (2021)
11. A. Sorini, E. J. Pineda, J. Stuckner et al., A convolutional neural network for multiscale modeling of composite materials. AIAA Scitech 2021 Forum. 0310, (2021)
12. R. Haj-Ali, J. Aboudi, A new and general formulation of the parametric HFGMC micromechanical method for two and three-dimensional multi-phase composites. Int. J. Solids Struct. **50**(6), 907–919 (2013)
13. H.-C. Shin, H.R. Roth, M. Gao et al., Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning. IEEE Trans. Med. Imaging. **35**(5), 1285–1298 (2016)
14. J. Jung, Y. Kim, J. Park et al., Transfer learning for enhancing the homogenization-theory-based prediction of elasto-plastic response of particle/short fiber-reinforced composites. Compos. Struct. **285**, 115210 (2022)
15. N.J. Nagelkerke, A note on a general definition of the coefficient of determination. Biometrika **78**(3), 691–692 (1991)
16. T. Kumar, K. Verma, A theory based on conversion of RGB image to gray image. Int. J. Comput. Appl. **7**(2), 7–10 (2010)
17. W. Oh, B. Lindquist, Image thresholding by indicator kriging. IEEE Trans. Pattern Anal. Mach. Intell. **21**(7), 590–602 (1999)
18. N. Jiang, L. Wang, Quantum image scaling using nearest neighbor interpolation. Quantum Inf. Process **14**(5), 1559–1571 (2015)
19. L. Al Shalabi, Z. Shaaban, B. Kasasbeh, Data mining: A preprocessing engine. J. Comput. Sci. **2**(9), 735–739 (2006)
20. Z. Yang, Y.C. Yabansu, R. Al-Bahrani et al., Deep learning approaches for mining structure-property linkages in high contrast composites from simulation datasets. Comput. Mater. Sci. **151**, 278–287 (2018)
21. Z. Li, F. Liu, W. Yang et al., A survey of convolutional neural networks: analysis, applications, and prospects. IEEE Trans. Neural Netw. Learn Syst. 1–21 (2021)
22. J. Nagi, F. Ducatelle, G. A. Di Caro et al., Max-pooling convolutional neural networks for vision-based hand gesture recognition. in *2011 IEEE Int. Conf. Signal Image Processing Appl.* 342–347, (2011)
23. N. Srivastava, G. Hinton, A. Krizhevsky et al., Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. **15**(1), 1929–1958 (2014)
24. C.J. Willmott, K. Matsuura, Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. Clim. Res. **30**(1), 79–82 (2005)
25. D. P. Kingma, J. Ba, Adam: A method for stochastic optimization. *arXiv preprint* arXiv:1412.6980*, (2014)*
26. W. An, H. Wang, Y. Zhang et al., Exponential decay sine wave learning rate for fast deep neural network training. in *2017 J Vis Commun Image Represent.* 1–4, (2017)
27. A. De Myttenaere, B. Golden, B. Le Grand et al., Mean absolute percentage error for regression models. Neurocomputing **192**, 38–48 (2016)

28. J. Seuffert, L. Bittrich, L. Cardoso de Oliveira et al., Micro-scale permeability characterization of carbon fiber composites using micrograph volume elements. Front. Mater. 428, (2021)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.