RESEARCH PAPER

A Lightweight Damage Diagnosis Method for Frame Structure Based on SGNet Model

C. Cai[1] · W. Fu[1] · X. Guo[1] · D. Wu[2] · J. Ren[1]

Abstract

Due to the complex structure of most frame structure, a large amount of sensor data needs to be processed for damage diagnosis, which increases the computational cost of diagnosis models and poses a serious challenge to their fast, accurate, and efficient damage diagnosis. In order to address this issue, this paper proposes a novel lightweight damage diagnosis method of frame structure for mobile devices based on convolutional neural networks. This method first uses mean filtering to process the vibration data collected by sensors, and then innovatively combines two convolutional neural network models, ShuffleNet and GhostNet, to form a new lightweight convolutional neural network model called SGNet, thereby reducing the computational cost of the model while ensuring diagnosis accuracy. In order to test the performance of the method proposed in this article, experimental research on damage degree diagnosis and damage type diagnosis is conducted by taking the frame structure provided by Columbia University as the research object, and comparative experiments of performance are conducted with MobileNet, GhostNet, and ShuffleNet. The experimental results show that the lightweight damage diagnosis method for frame structure proposed in this article not only has high damage diagnosis accuracy, but also has fewer computational parameters, when the highest accuracy is 99.8%, the computational parameters are only 1 million. At the same time, it is superior to MobileNet, GhostNet, ShuffleNet in terms of diagnosis accuracy and computational cost, so it is an effective high-precision lightweight damage diagnosis method for frame structure.

Keywords Frame structure · Mean filtering · Lightweight model · ShuffleNet · GhostNet

Introduction

With the advancement of science and technology, modern productivity has significantly improved, and the application of frame structure has become widespread in various fields, including mining machinery, civil engineering, aerospace, and bridge construction [1, 2]. Frame structure consists of interconnected members held together by bolts or welding, it often experiences failures due to factors such as bolt loosening, uneven force distribution and oxidation [3]. These failures may lead to machinery malfunction and catastrophic collapse of the frame structure, which can pose significant risks to human life, property, and safety. Therefore, it is

of great engineering practical significance to propose effective damage diagnosis methods for state detection and damage identification of frame structure, and to make early predictions of their healthy operating states.

The composition of the frame structure is increasingly moving towards gigantism, complexity and modularity. However, the rising computational costs of data pose challenges in achieving effective damage diagnosis for these structures. Traditional damage diagnosis methods include short time Fourier transform (STFT) [4], K − nearest neighbor algorithm (KNNA) [5, 6], fuzzy cluster analysis (FCA) [7, 8] and peak to peak comparison (PTPC) [9]. Li et al. conducted numerical research on planar truss structures by using autocorrelation functions of structural acceleration responses under white noise excitation to form a covariance matrix; they identified damage conditions under different noise levels [10]. Malekjafarian et al. proposed an improved transition mode identification method by using Hankel matrix averaging to detect closely spaced modes [11]. Yang et al. employed empirical mode decomposition (EMD) and

✉ D. Wu
  wudiwudi@hebeu.edu.cn

1 School of Mechanical and Equipment Engineering, Hebei University of Engineering, Handan 056038, China

2 School of Information and Electrical Engineering, Hebei University of Engineering, Handan 056038, China

Hilbert transform to extract damage peaks caused by sudden changes in structural stiffness, thereby achieving detection of the moment and location of damage occurrence [12]. Li et al. investigated a combined method by using EMD and wavelet analysis to detect changes in structural response data; they decomposed the structural vibration response signal into multiple single−component signals by using EMD, and then transformed them into analytical signals through the Hilbert transform; subsequently, they performed a wavelet transform on each single−component signal to accurately identify damage location and severity [13]. Zhu et al. proposed a bearing fault diagnosis method based on wavelet packet decomposition and KNNA; this method first decomposed the original bearing vibration signal by using wavelet packet decomposition, then calculated the sample entropy value for each decomposed signal to construct a feature vector, and finally employed KNNA for bearing fault diagnosis [14]. Despite the widespread application of traditional damage diagnosis methods in various fields, the changing complexity of frame structure with scientific progress poses challenges. When using traditional damage diagnosis methods to deal with damage diagnosis problems with complex damage mechanisms, numerous classification categories and massive data, it may lead to a decrease in diagnostic performance. This problem poses a challenge to achieving efficient and accurate damage diagnosis, which contradicts the requirements for rapid and intelligent development of structural damage diagnosis.

In recent years, with technological advancements and continuous algorithms improvements, deep learning has achieved significant success in various fields. For instance, in computer vision [15–17], natural language processing [18, 19], speech recognition [20, 21], and autonomous driving [22, 23]. Kostic et al. combined sensor clustering−based time series analysis with artificial neural networks for bridge damage detection under temperature variations; they performed 2000 simulations with temperature effects and damage conditions by using a pedestrian bridge finite element model [24]. Khodabandehlou et al. utilized vibration signals and a two−dimensional deep convolutional neural network to extract features from historical acceleration responses and reduce the dimensionality of the response history; this enabled damage state classification through a limited number of acceleration measurements [25]. Avci et al. proposed a one−dimensional convolutional neural network−based wireless sensor network (WSN) for real−time and wireless structural health monitoring; in this method, each CNN was assigned to its local sensor data, and the respective models were trained for each sensor unit without any synchronization or data transmission [26]. Tang et al. segmented the original time series data and applied visual processing in both time and frequency domains; then they overlaid these segmented images into single or double−channel images and labeled them based on visual features; subsequently, they designed and trained a CNN for data anomaly classification [27]. Cuşkun et al. employed a novel 3D deep learning architecture to classify MR images of patients with brain tumors, thereby determining the primary site of brain metastasis. [28]. Al-Areqi and colleagues proposed a machine learning approach for the rapid diagnosis of the Covid-19 disease, with a focus on the impact of different features on classification accuracy [29]. Yue et al. proposed a fault diagnosis method by using deep adversarial transfer learning; they used a single−layer CNN and transferred learning to employ the ResNet residual network as both the generator and discriminator in a GAN and obtained higher accuracy in both GAN recognition and generation capabilities [30]. GAN has shown high accuracy in dealing with problems with limited training samples and can effectively extract feature information even from one−dimensional vibration data. However, GAN generates many simulated models, making training time−consuming and computationally expensive. It exhibits unique accuracy advantages when handling problems with diverse vibration data but limited sensor numbers. However, in the context of fault diagnosis for frame structure with a large amount of data and numerous sensors, it often faces drawbacks such as low efficiency, limited diagnostic accuracy and poor intuitiveness.

In order to solve the problem of accuracy degradation caused by multi-sensor data in frame structure damage diagnosis and reduce the computational cost of the model, and achieve accurate damage diagnosis on mobile devices, this paper proposed a new structural damage diagnosis method. Firstly, the sensor data was subjected to mean filtering to achieve smoother data. Subsequently, the processed data was input into the SGNet model for training. The foundation of the SGNet model is based on the ShuffleNet [31] and GhostNet [32], they are lightweight models. By making appropriate improvements to these models, the new SGNet model became more suitable for structural damage diagnosis in building frame structures, thereby enhancing the efficiency and accuracy of frame structure diagnosis. This article has two important contributions. One is to propose an accurate damage diagnosis method for frame structure in a multi-sensor data environment, and the other is to propose a lightweight structural damage diagnosis model suitable for mobile devices while ensuring the accuracy of damage diagnosis, greatly reducing the computational cost of the model.

This paper consists of 6 parts, "Signal Filtering Method" section introduces the signal filtering method, "Neural Network Model" section presents the neural network model, and "Damage Diagnosis Process" section introduces the damage diagnosis process. "Experimental Study" section is the main content of the experiment, including experimental

objects, experimental data, damage degree diagnosis experiment, damage type diagnosis experiment, model comparison experiment and discussion. "Conclusion" section summarizes this paper and draws relevant conclusions.

Signal Filtering Method

Filtering is a commonly used method in signal processing; it is used to remove noise or unwanted components from signals while preserving helpful information. Noise arises from random fluctuations caused by measurement errors, sensor interference, or other environmental factors. Filtering aims to extract useful information from signals while suppress or eliminate redundancies. Mean filtering and median filtering are two standard methods used for this purpose.

Mean filtering is a linear filtering method that can achieve signal smoothing by replacing each sample point with the average value of samples in its surrounding neighborhood. The mathematical formula for mean filtering can be represented as:

$$y[n] = \frac{1}{N} \sum_{i=n-N/2}^{n+N/2} x[i] \tag{1}$$

where $y[n]$ is the filtered signal sample, $x[i]$ is the original signal sample, and N determines the size of the neighborhood used for calculating the average value. A larger neighborhood size can provide stronger smoothing effects but may result in the loss of signal details.

Median filtering, on the other hand, is a non−linear filtering method that can remove noise by replacing each sample point with the median value of samples in its surrounding neighborhood. The mathematical formula for median filtering can be represented as:

$$y[n] = \text{median}\left(x[n - N/2], x[n - N/2 + 1], \ldots, x[n + N/2]\right) x[i] \tag{2}$$

where $y[n]$ is the filtered signal sample, $x[i]$ is the original signal sample, and N determines the size of the neighborhood used for calculating the median value. Median filtering

is suitable for cases where noise statistics do not follow the Gaussian distribution and isolated outlier values are present. A larger neighborhood size can remove larger−size noise, but it may lead to blurring of signal.

Using mean and median filtering as signal processing methods can improve signal quality. However, their applicability depends on specific application cases and the statistical characteristics of the noise. In some cases, these methods can improve accuracy, but in others, they may impact signal details. Therefore, adjustments and evaluations should be made based on the specific problem when using these methods. Generally, sampling mean filtering is often used when the signal follows the Gaussian distribution. On the other hand, if the signal does not follow the Gaussian distribution and the preservation of signal edges and detail features is a concern, median filtering is more suitable.

Neural Network Model

ShuffleNet Model

ShuffleNet is a lightweight convolutional neural network model proposed by Megvii in 2018. Its main features are group convolution, channel shuffle and depthwise separable convolution. Group convolution divides the input tensor into multiple subgroups and performs independent convolution operations on each subgroup, which can reduce computational complexity and model parameters. Channel shuffle rearranges the output tensor of group convolution to achieve cross−group information fusion and reduce information bottlenecks; it is shown in Fig. 1. Depthwise separable convolution is a lightweight convolutional operation that splits the standard convolution into depth and point−wise convolution, it can further reduce computational complexity and model parameters. Compared to the MobileNet [33] architecture, ShuffleNet demonstrates significant advantages in terms of performance; it has smaller parameters and computational
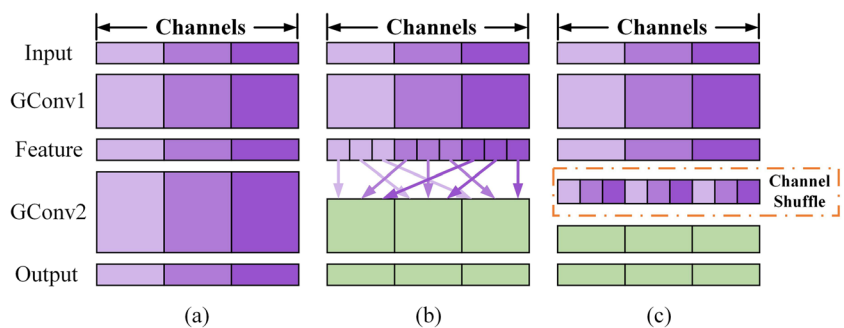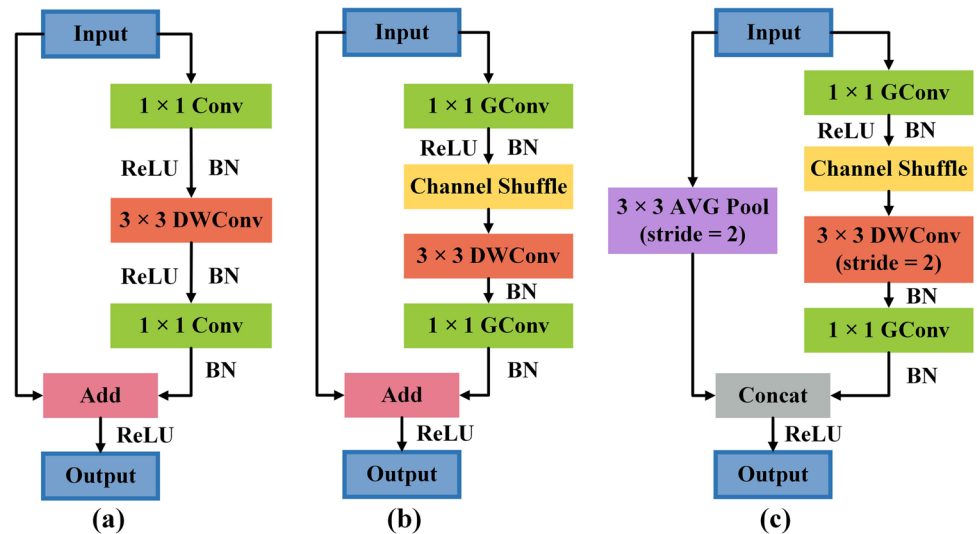
Fig. 1 Channel Shuffle structure

Fig. 2 ShuffleNet unit structure


(a) **(b)** **(c)**

sizes, and higher accuracy. ShuffleNet is mainly composed of multi-layer ShuffleNet unit structures, and ShuffleNet unit mainly utilizes the advantages of channel rearrangement and combines the residual principle of ResNet [34] model, it is illustrated in Fig. 2.

GhostNet Model

The GhostNet model is a lightweight convolutional neural network model proposed by Huawei Noah's Ark Lab in 2019. Deep convolutional neural networks [35, 36] typically consist of many convolutions, resulting in a significant increase in computational costs. In contrast, the GhostNet model has a relatively simple network structure, enabling faster training and inference speeds with smaller computational and parameter sizes while achieving higher accuracy. Its distinguishing feature is the introduction of the Ghost Module structure.

The Ghost Module is a lightweight convolutional module proposed to extend ordinary convolutions. It achieves this by splitting the input channels into two parts: the main branch and the ghost branch. The convolution kernels of the main and ghost branches are independent. The output of the main branch serves as the output of the entire module. In contrast, the production of the ghost branch can be discarded or used for subsequent operations, thus reducing computational costs; it is shown in Fig. 3.

GhostNet comprises multiple Ghost Bottlenecks; each Ghost Bottleneck is formed by stacking multiple Ghost Modules. When the stride is 1, a Ghost Bottleneck consists of two stacked Ghost Modules, which are connected by using residual connections. The first Ghost Module acts as an expansion layer to increase the number of channels,
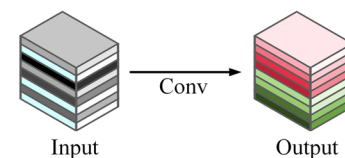
(a) Ordinary Convolution
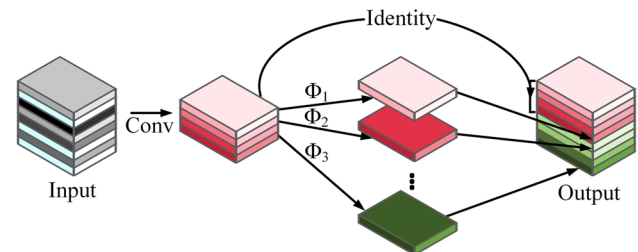
(b) Ghost Module

Fig. 3 Ghost Module structure

while the second Ghost Module reduces the number of channels to match the residual connection; it is shown in Fig. 4(a). When the stride is 2, in addition to Fig. 4(a) configurations, a 3×3 depthwise separable convolution is inserted between the two Ghost Modules, as illustrated in Fig. 4(b).

SGNet Model

Since both ShuffleNet and GhostNet are designed for lightweight and efficient models while maintaining accuracy and precision, combining the advantages of two models can result in a superior model ensemble effect. The different characteristics and advantages of ShuffleNet and GhostNet complement each other, providing more comprehensive

Fig. 4 Ghost Bottleneck structure

(a) Stride=1 Ghost Bottleneck   **(b)Stride=2 Ghost Bottleneck**
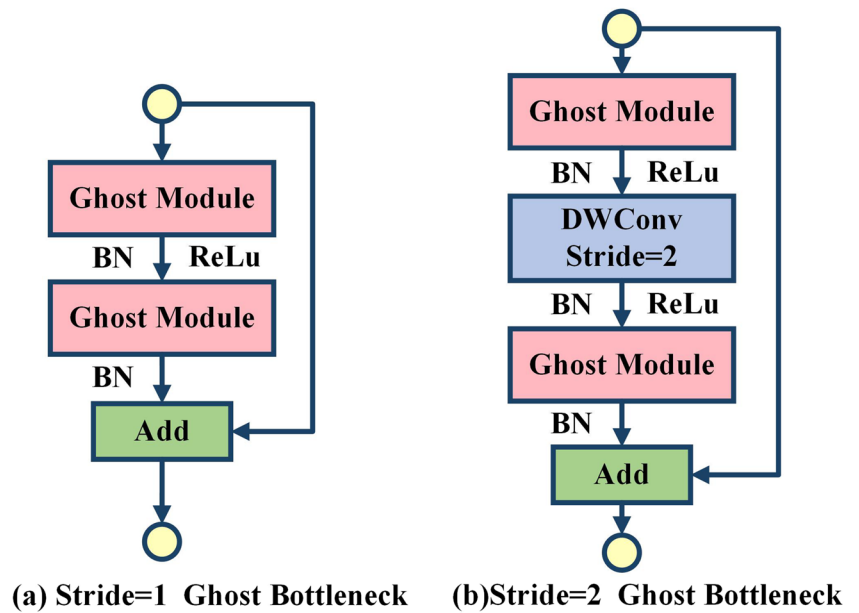
feature extraction and representation capabilities. ShuffleNet's channel shuffle and group convolution operations can help capture spatial information and feature correlations, while GhostNet's ghost channels can help improve parameter efficiency and feature utilization. Combining the advantages of the above two models not only ensures powerful and efficient feature extraction and model representation capabilities, but also enables a more lightweight model. The new model is named SGNet (ShuffleNet and GhostNet−based Network), and its structure is shown in Fig. 5.

Using large convolution kernels and strides can increase the receptive field and capture a larger range of features in the input signal, which helps extract vibration features relevant to damage. In addition, it can also reduce the size of the output feature maps, achieve signal downsampling and reduce computational costs and data dimensions. Therefore, the SGNet
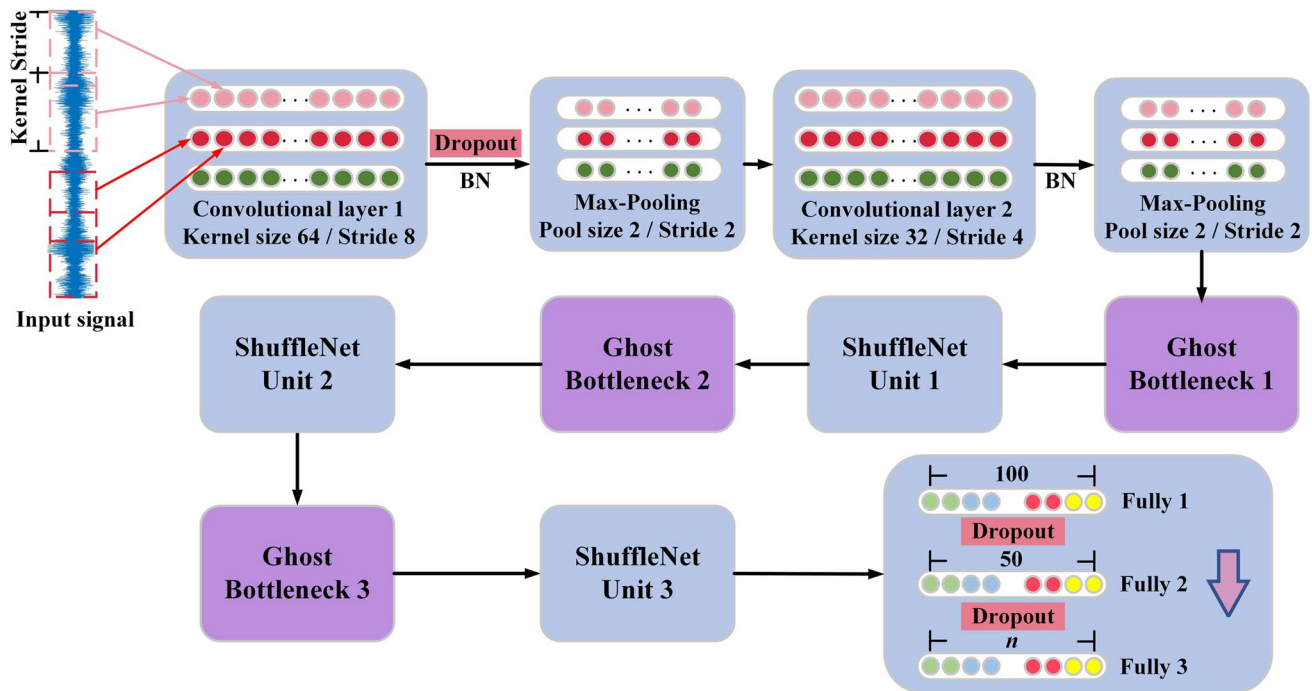
Fig. 5 SGNet model

model's initial layers utilize large convolution kernels and strides. The first convolutional layer has a kernel size of 64 and a stride of 8, followed by a Dropout layer. The second convolutional layer has a kernel size of 32 and a stride of 4. After each convolutional layer, Batch Normalization (BN) and max−pooling layers are applied. Following the second max−pooling layer, a structure with 3 Ghost Bottlenecks and 3 ShuffleNet units alternately distributed. In the latter part of the model, three fully connected layers are employed to obtain the output results. The number of outputs can be modified based on the classification categories, denoted as n. Dropout is added between each fully connected layer to prevent overfitting.

Damage Diagnosis Process

The damage diagnosis process mainly consists of three parts: data processing, model building and training, and model testing, as illustrated in Fig. 6.

(1) Data processing

Data processing consists of several steps, including data acquisition, mean filtering, data augmentation, data normalization and data partition. Firstly, vibration signals obtained from acceleration sensors under different conditions were denoted as S_{ij} (i represents the condition, j represents the sensor position), it shown in formula (3). Then, the acquired vibration signals underwent mean filtering, and the filtered data was organized and stored according to the sensor ID in data storage files.

$$S_{ij} = \begin{bmatrix} S_{11} & S_{12} & \cdots & S_{1j} \\ S_{21} & S_{22} & \cdots & S_{2j} \\ \vdots & \vdots & \vdots & \vdots \\ S_{i1} & S_{i2} & \cdots & S_{ij} \end{bmatrix} \tag{3}$$

Since a large amount of data is required for model training, but the acquired data samples are limited, data augmentation needs to perform to increase the number of data samples. Data augmentation helps avoid overfitting due to a small dataset and allows the model to learn the data distribution better, thereby enhancing the model's generalization ability. Sliding window overlapping sampling is a commonly used data augmentation method. The data after data augmentation was denoted as SA_{ij}, as shown in formula (4). The sliding window overlapping sampling can be depicted in formula (5), in which L represents the length of the vibration signal, W is the sliding window size, and S is the step size for sliding the window. N represents the final number of samples obtained.

$$SA_{ij} = \begin{bmatrix} SA_{11} & SA_{12} & \cdots & SA_{1j} \\ SA_{21} & SA_{22} & \cdots & SA_{2j} \\ \vdots & \vdots & \vdots & \vdots \\ SA_{i1} & SA_{i2} & \cdots & SA_{ij} \end{bmatrix} \tag{4}$$

$$N = \frac{L - W}{S} + 1 \tag{5}$$

Data normalization scales all feature information within a specified range, which is beneficial for the convolutional neural network to extract features from

Fig. 6 Damage diagnosis process
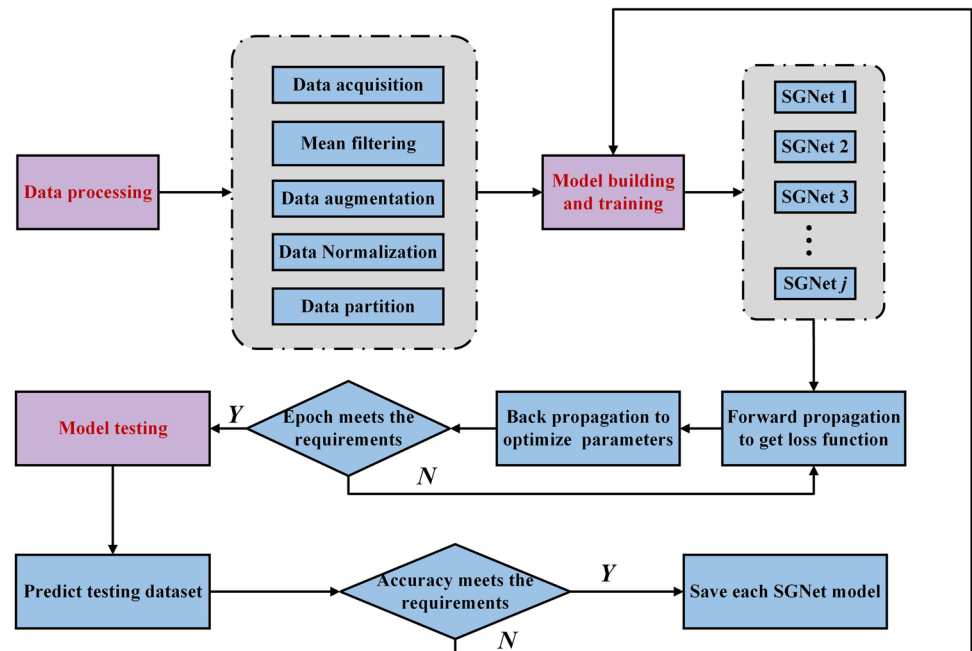
signals, thus improving the algorithm's stability, convergence speed and accuracy. The normalized data was represented as SAN_{ij}, as shown in formula (6). After normalization, the data was divided into training dataset, validation dataset and testing dataset based on the corresponding proportions.

$$SAN_{ij} = \begin{bmatrix} SAN_{11} & SAN_{12} & \cdots & SAN_{1j} \\ SAN_{21} & SAN_{22} & \cdots & SAN_{2j} \\ \vdots & \vdots & \vdots & \vdots \\ SAN_{i1} & SAN_{i2} & \cdots & SAN_{ij} \end{bmatrix} \quad (6)$$

(2)　Model building and training

For model training, the model structure needs to be constructed first. The number of required models was determined based on the number of sensors for detection. Then, the model underwent forward propagation during training to obtain the output results and calculate the loss function. After getting the loss function, the model performed backward propagation for updating, computing gradients and optimizing parameters. As training epochs increased, the accuracy of the model can be continuously improved while the loss function will continue to decrease until the model reaches the set number of training epochs.

(3)　Model testing

After SGNet j reached the designated number of training epochs, the testing dataset of sensor j was input into the model. The model parameters must be modified and retrained if the output accuracy does not meet the requirements. If the output accuracy meets the requirements, the model can be saved.

Experimental Study

This section includes the experimental object, experimental data, damage degree diagnosis experiment, damage type diagnosis experiment, model comparison experiment and discussion. The deep learning framework used in the experiments is Tensorflow version 2.6.1, and the computer CPU used in the experiments is Intel Core i7−10750H with an NVIDIA GTX 1660Ti GPU.

Experimental Object

The experimental object used in this study is a four−story frame structure constructed by Columbia University [37]. Its 3D model is shown in Fig. 7(a). The size of the frame structure base is 2.5 m × 2.5 m, and the height of the frame structure is 3.6 m. The structure consists of four faces: east, south, west, and north, each face is composed of beams and columns with the same structural size. Different components of various orientations were represented by the same codes (e.g., east 1, north 1) in the fig. A total of 15 acceleration sensors were placed on the frame structure. Each floor junction had 3 acceleration sensors (1 on the west, 1 on the east, and 1 near the central column). Acceleration sensors numbered 1–3 were placed at the ground level, while the rest were positioned at the relevant locations on each floor's top. The positions of the sensors are shown in Fig. 7(b).

By removing or loosening the diagonal bracing and bolt connections numbered 1–12 in Fig. 7(a), nine different damage cases were simulated in the experiment, and the specific operations for each case are shown in Table 1. Since the overall damage degree of the frame structure varies in these nine cases, there are significant differences between the collected vibration signal data. Therefore, the differences in the data can be used to distinguish the damage cases of the frame structure.

Fig. 7 Frame structure and sensor position [37]

(a) 3D structure　　　　　　　　　　(b) Sensor position
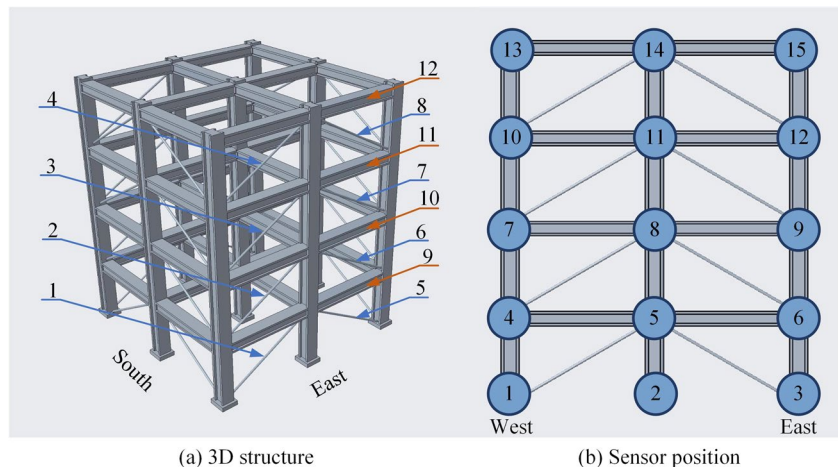
Table 1 Specific damage cases

| Cases	Operation
1	Undamaged (intact)
2	Removal of east face component 1
3	Removal of east face components 1 and 4
4	Removal of east face components 1–4
5	Removal of east face components 1–8
6	Removal of east face components 1–8 and north face components 2 and 6
7	Removal of components 1–8 from all four faces
8	Removal of components 1–8 from all four faces and loosening of components 9 and 10 on the east face
9	Removal of components 1–8 from all four faces and loosening of components 9–12 on the east face

Fig. 8 Signal filtering results
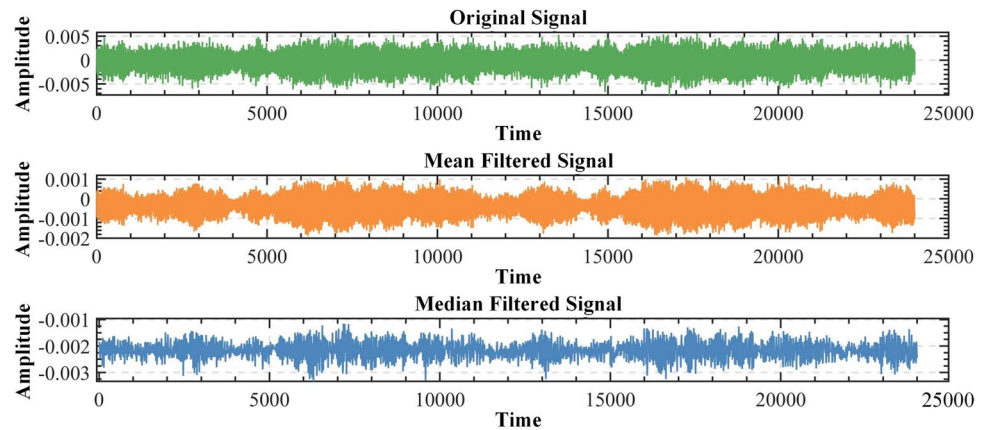
Experimental Data

In experiments, the frame structure was sequentially damaged according to the Table 1, and a 200 Hz impact was applied to the frame structure to acquire vibration signals. Then, the vibration signals were processed by using the data processing procedures in "Damage Diagnosis Process" section.

From the publicly available data provided by Columbia University, 135 data were obtained from the 15 sensors under the nine damage cases. In experiments, both mean filtering and median filtering were used to process the data. The processed results of the data obtained from the sensor numbered 13 in case 1 are shown in Fig. 8. From the figure, it can be observed that after using mean filtering, the data edges become smoother, and after using median filtering, the data density becomes smaller with prominent edges.

To select an appropriate filtering method, it is necessary to further determine whether the data conforms to the Gaussian distribution. Taking the data of the sensor numbered 5 under case 1 as an example; the histogram, Quantile−Quantile (Q−Q) plot, skewness, and kurtosis of the data were plotted in Fig. 9. From the figure, it can
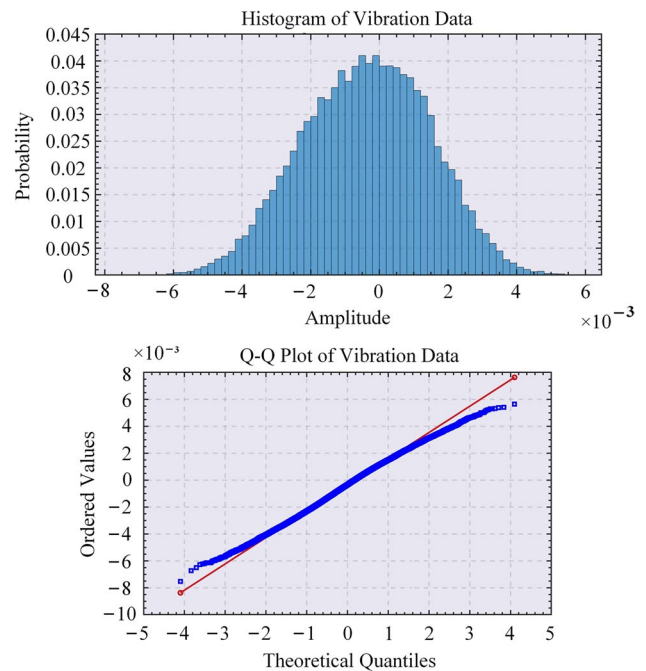
Fig. 9 Signal histogram and Q-Q plot

be observed that the histogram shape is approximately bell−shaped, the data points in the Q − Q plot are distributed close to the straight line, and the skewness of the signal is 0.007, with a kurtosis close to 3.49, this indicates that the signal approximately follows the Gaussian distribution. As mentioned in "Signal Filtering Method" section, when the data follows the Gaussian distribution, the mean filtering method is more suitable. Therefore, based on the analysis results from Figs. 8 and 9, the mean filtering method was selected to process the data in experiments.

The length of the data collected in the experiment varies under different cases, with the lengths of 24,000, 60,000, and 72,000 for cases 1–5, case 6 and cases 7–9, respectively. Different sizes of sliding step were used for different cases to ensure the number of samples was similar for each case during data augmentation. The sliding window size W was set to 1024 for all cases, and the sliding step sizes S for cases 1–5, case 6, and cases 7–9 were set to 4, 10, and 12, respectively. After enhanced processing of the data, the number of samples N for cases 1–5, case 6, and cases 7–9 were 5744, 5898, and 5915, respectively.

Damage Degree Diagnosis Experiment

Since case 1 and case 9 represented the undamaged and damaged conditions of the frame structure, respectively, the data (SAN_{1j} and SAN_{9j}) of the sensor numbered j under case 1 and case 9 were used for binary classification training in the SGNet j model ($j = 1$–15). The SGNet j model with the required accuracy were saved. Then, the data of each sensor under different cases was input into each model. In other words, for each case i and the sensor numbered j, the data was input to model j to obtain the diagnosis result. Based on the obtained results, the Pod (Probability of damage) was calculated according to Fig. 10. Finally, the average probability Pod_{avg} of all sensors under case i represented the damage degree of the frame structure under case i. The larger the Pod_{avg} is, the more severe the overall damage to the frame structure.

Training Results Analysis

Due to the extensive number of sensors used in experiments, this section primarily focused on training results with data collected from the sensor numbered 13. In order to study the impact of different parameters on the model, this paper used convolution kernels of different sizes and strides for training. During training, the Adam optimizer and cross-entropy loss function were employed. The experimental results were presented in Fig. 11, in which the notation "First 32-8 Second 16-4" signifies that the first convolutional layer used kernel
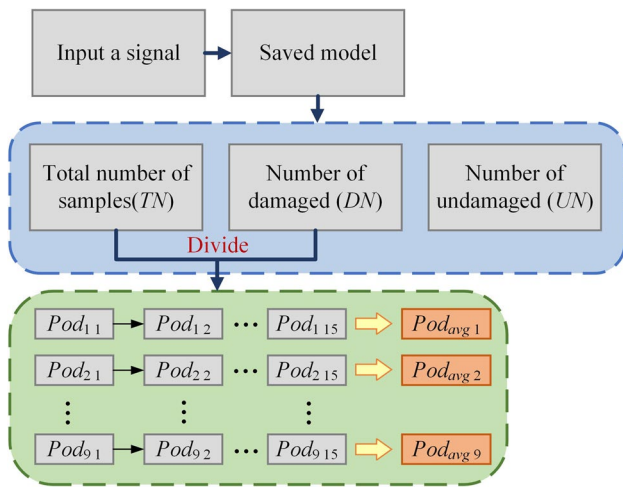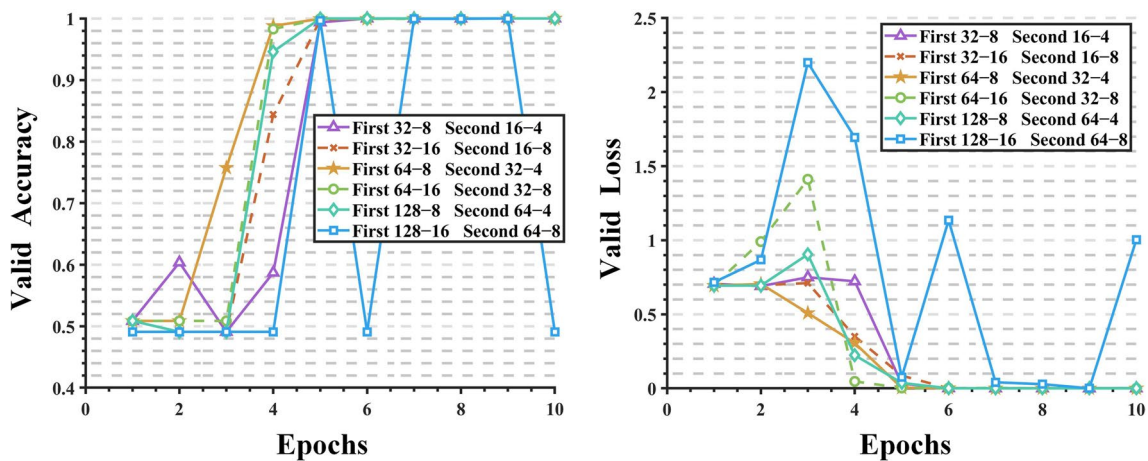

Fig. 10 The calculation process of *Pod*

Fig. 11 The influence of different convolution kernels and strides on the model

sizes of 32 and strides of 8, while the second convolutional layer used kernel sizes of 16 and strides of 4.

It can be seen from Fig. 11 that, under the "First 64-8 Second 32-4" configuration, the validation accuracy exhibits the most significant improvement, the loss function decreases rapidly, and the training process remains stable. Conversely, under the "First 128-16 Second 64-8" configuration, there is substantial fluctuation in the training process. Excessive kernel sizes and strides can lead to a rapid reduction of feature maps, leading to the model ignoring important information in the data. Consequently, the model's parameters were determined to be "First 64–8 Second 32–4" based on comprehensive evaluation of the model's performance on the validation dataset.

Under the above selected parameters, the training process of model SGNet 13 for 30 epochs is shown in Fig. 12. From the figure, it can be observed that the model starts to converge when the training epoch reaches 5. The training and validation datasets' accuracy exceeds 99%, and the loss is below 1×10^{-4}. When the training epoch reaches 25, the loss of the training dataset stabilizes around 5×10^{-4}, and the validation dataset's loss remains close to 0.

Confusion Matrix Analysis

The confusion matrix, also known as the error matrix, is a method used to evaluate the performance of a model. It shows the relationship between the model's predicted and true labels. By analyzing the confusion matrix, classification indicators such as accuracy, recall, precision, and F1 score can be calculated, which can comprehensively evaluate the performance and error types of the classification model, help understand the classification performance of the model, and further adjust and optimize the model as needed.

The confusion matrices of the training, validation, and testing datasets for model SGNet 13 during different training epochs are shown in Fig. 13. By calculating the data in the figure, it can be seen that when the training epoch reaches 4, the training, validation, and testing dataset's accuracy is approximately 86%. The recall is 0.787, the precision is 1,

and the F1 score is 0.881. When the training epoch reaches 30, the accuracy, recall, precision and F1 score for all three datasets reach a perfect score of 1. The model can accurately distinguish between undamaged and damaged data.

Damage Degree Diagnosis Results

Based on the damage probability calculation process shown in Fig. 10, the Pod_{ij} and Pod_{avg} for 15 sensors under 9 different damage cases were recorded in the experiment. The data was presented in Table 2 and visualized in Fig. 14.

In Case 1, the Pod for all 15 sensors is below 1%. In Case 2, the Pod of each sensor is below 10%. In Case 3, the Pod of the sensor numbered 2 is the highest, it is 99.48%. In Case 4, the Pod of sensors numbered 2 and 4 are above 80%, while the rest are below 65%. In Cases 5–6, the highest Pod of sensors numbered 4, 6, 9, 12, and 15 can reach 94%. In Cases 7–9, the highest Pod of sensors numbered 4–15 can reach 100%. When using data from sensors numbered 5, 8, 11, and 14 at the center column for damage diagnosis, the Pod of cases 1–5 does not change much. As the degree of damage to the frame structure increases, Pod_{avg} also gradually increases, and there is a certain difference between each Pod_{avg}. Therefore, the overall damage degree of the frame structure can be judged based on the size of Pod_{avg}. The proposed SGNet model performs excellently in damage degree diagnosis of frame structure and can be used to determine the damage degree of frame structure and further identify the damaged locations.

Damage Type Diagnosis Experiment

The purpose of the damage type diagnosis experiment is to conduct multi classification training on the model, detect whether the model can distinguish the damage case where the sensor's data belongs, and test the classification ability of the model. In this experiment, a new SGNet model was constructed, and data from each sensor under 9 different cases were used for training to obtain a 9−class classification

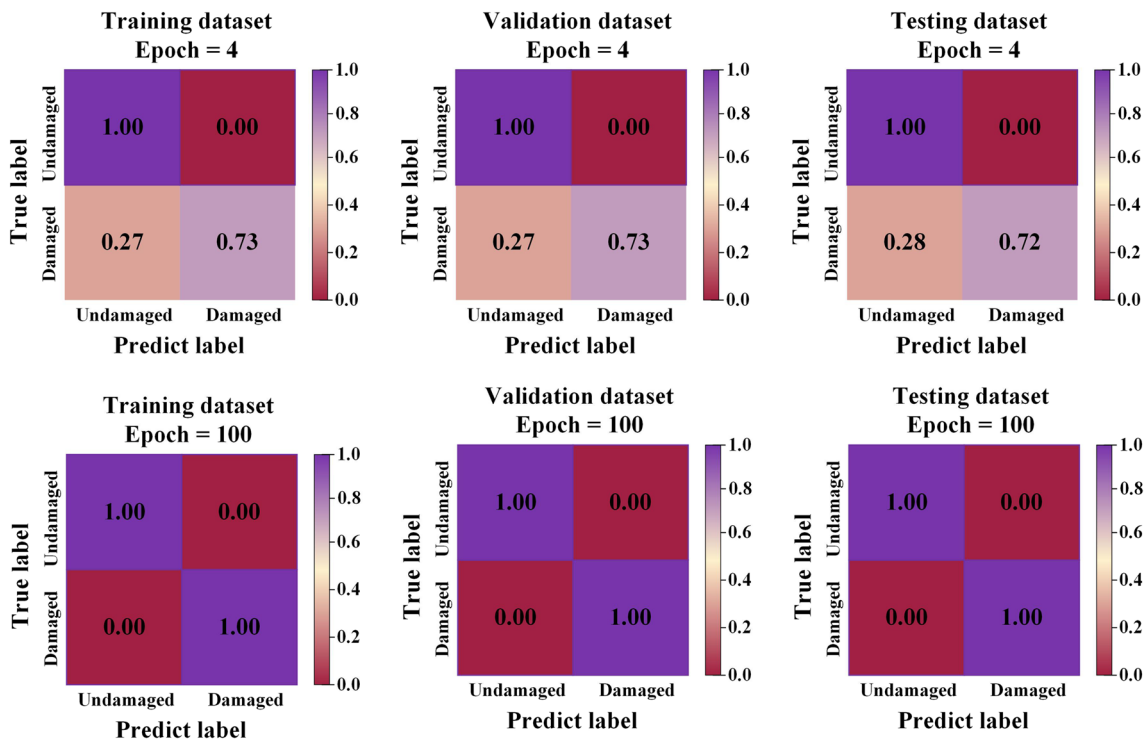**Fig. 12** Training results of damage degree
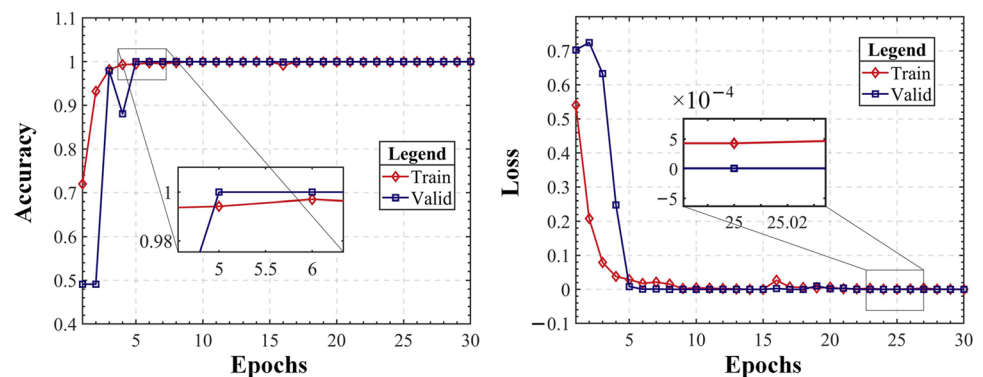
Fig. 13 Confusion matrix of damage degree experiment

Table 2 Pod_{ij} and Pod_{avg} for different sensors and cases

	Case1	Case 2	Case 3	Case 4	Case 5	Case 6	Case 7	Case 8	Case9
1	0.10%	0%	0%	5.74%	13.78%	0.10%	99.39%	99.90%	100%
2	0%	0%	99.48%	81.42%	79.02%	0.51%	9.74%	100%	100%
3	0%	4.38%	0%	0%	6.26%	0.51%	4.16%	77.18%	100%
4	0%	4.80%	25.78%	80.69%	87.03%	94.25%	99.49%	100%	100%
5	0%	0%	0%	0%	0%	1.53%	100.00%	100%	100%
6	0%	0%	0.31%	0%	70.52%	88.12%	94.08%	99.90%	100%
7	0%	0.21%	14.09%	36.10%	52.82%	65.21%	65.21%	100%	100%
8	0%	0%	0%	0%	0.84%	0.10%	100.00%	100%	100%
9	0%	7.93%	72.32%	62.08%	80.51%	89.16%	93.31%	98.28%	100%
10	0%	0%	4.38%	1.98%	8.66%	22.18%	99.80%	100%	100%
11	0%	0%	0%	0%	0%	78.08%	62.14%	84.38%	100%
12	0%	10.13%	15.97%	41.96%	92.90%	76.20%	100%	100%	100%
13	0%	1.46%	10.65%	61.69%	8.14%	22.08%	90.47%	100%	100%
14	0%	0%	0.31%	0%	0%	0.10%	100%	100%	100%
15	0%	0%	43.11%	34.86%	86.53%	91.27%	95.74%	100%	100%
Avg1–15	0.01%	1.93%	19.09%	27.10%	39.13%	41.96%	80.90%	97.30%	100%
Avg4–15	0%	2.04%	15.58%	26.61%	40.66%	52.36%	91.69%	98.54%	100%

result. On this basis, the performance of the model was tested through training results, and the damage types of the frame structure were obtained.

Training Results Analysis

Taking the training results of the data collected by sensor numbered 13 as an example; the results are shown in Fig. 15. The figure shows that when the training epoch reaches 3, the accuracy of the training dataset is above
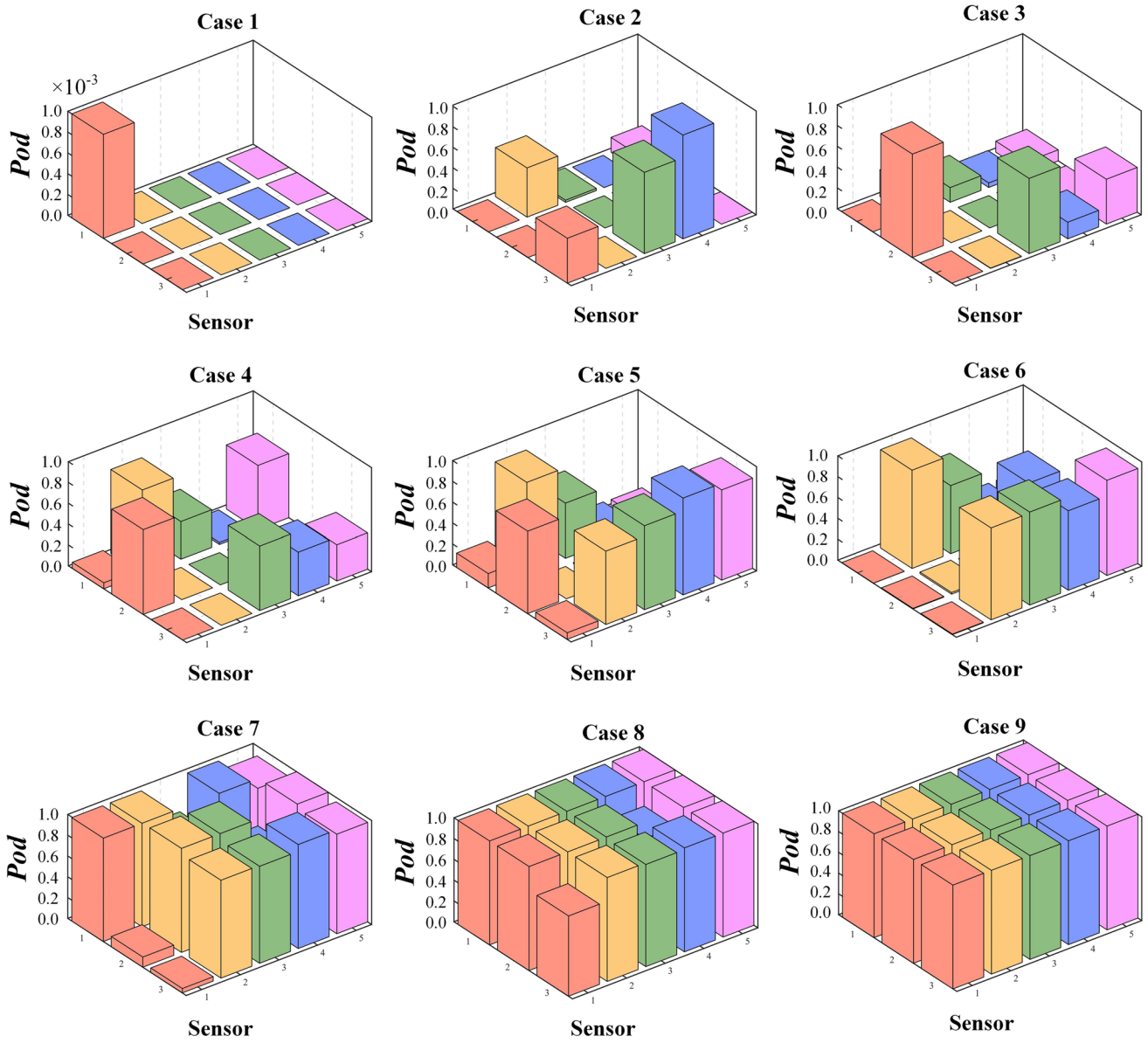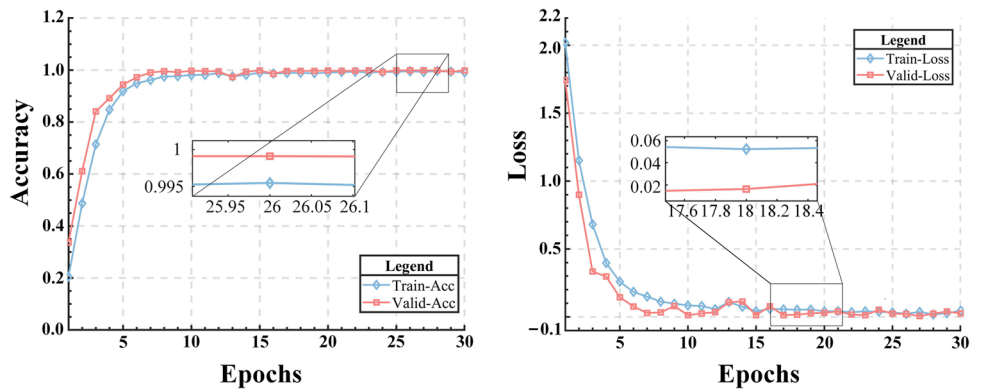
Fig. 14 Pod_{ij} for different sensors and cases

Fig. 15 Training results of damage type experiment

70%, while the accuracy of the validation dataset is 84%; the loss of the training dataset is around 0.7, and the loss of the validation dataset is 0.3. After the training epoch reaches 5, the training curves begin to converge; the accuracy reaches over 90%, and the loss decreases to below 0.1. When the training epoch reaches 26, the training and validation dataset's accuracy is higher than 99%, and the loss is around 0.02. From the above results, it can be seen that the SGNet model has a faster convergence speed, higher accuracy, and lower loss during the training process.

In the initial stages of training, the model's parameters typically start in a randomly initialized state. This leads to some degree of variability in training results at the outset, where the loss function and accuracy may exhibit instability. This is because the model needs to adapt to the data and gradually adjust its parameters for better fitting.

As the number of training epoch increases, the model gradually converges towards a state closer to the optimal solution. This is manifested in an incremental improvement in accuracy and a gradual decrease in the loss function. The model fine-tunes its parameters over time through the optimization algorithm to fit the training data more effectively.

When a certain stage is reached with an adequate number of training epochs, the model's parameters have essentially found the optimal solution. Consequently, the accuracy becomes very high, and the loss function is extremely low. This indicates that the model has become highly stable at this point and can perform tasks with a high degree of accuracy.

Confusion Matrix Analysis

Taking the training results of the data collected by sensor numbered 13 as an example; the confusion matrices are shown in Fig. 16. In the matrices, labels 1 to 9 represent the 9 damage cases. When the training epoch reaches 10, the accuracy of training, validation and testing is 99%, 98.9%, and 99%, respectively; when the training epoch reaches 30, the accuracies for all three datasets reach 100%. The above results indicate that the model has strong classification ability and achieved satisfactory classification results.

T − SNE Visualization

T − SNE (T − Distributed Stochastic Neighbor Embedding) is a machine learning algorithm for dimensionality reduction and data visualization in high−dimensional spaces. It can map high−dimensional data to two−dimensional or three−dimensional space, and effectively display the structure and relationships of high-dimensional data and reveal patterns and clusters in the data. T-SNE is of great value in

exploratory data analysis and visualization, as it can capture nonlinear relationships while preserving local structures, making the results easy to observe and understand.

Taking the visualization result of the data collected by sensor numbered 13 as an example, the data during the training process was reduced to lower dimensions by using T − SNE, and the visualization results of the original data, the data of the 10 training epoch, and the data of the 30 training epoch were obtained, it is shown in Fig. 17.

It can be seen from Fig. 17 that the original data shows a relatively scattered distribution of the 9 damage cases. When the training epoch reaches 10, the different cases can be somewhat distinguished and case 7 and case 9 have some dispersion at their edges. When the training epoch reaches 30, the model can completely distinguish different data labels.

Model Comparison Experiment

Model Parameter Quantity

The number of parameters in a model determines the diagnosis equipment and computational cost. MobileNet V1 has approximately 4.2 million parameters, GhostNet has about 5.2 million parameters, ShuffleNet 0.5x has about 1.8 million parameters, ShuffleNet 1.0x has about 2.3 million parameters, while SGNet has only about 1 million parameters, as shown in Table 3. SGNet has high flexibility and adjustability, and can be adjusted and optimized according to specific situations. From the experimental results, it can be seen that SGNet has smaller parameter quantities and lower computational costs compared to other classical models in terms of the total parameter quantity of the model.

Testing Dataset Accuracy

To verify the superiority of the SGNet model, this section compared the testing dataset accuracy of SGNet, MobileNet, GhostNet, and ShuffleNet under the same experimental conditions. The experiment data was taken from the damage type diagnosis experiment. The number of training epochs was set to a fixed value, and the accuracy and average accuracy of the test dataset for four models with training epochs of 1, 3, 5, 10, 15, 20, 25, and 30 were recorded. The results are shown in Table 4, and the data in Table 4 is visualized in Figs. 18 and 19.

The experimental results in Table 4 can demonstrate the following performance.

- When the training epoch reaches 1, the accuracy of SGNet is 15.1%, the accuracy of MobileNet and Ghost-Net is 10.9%, and the accuracy of ShuffleNet is 10.1%.
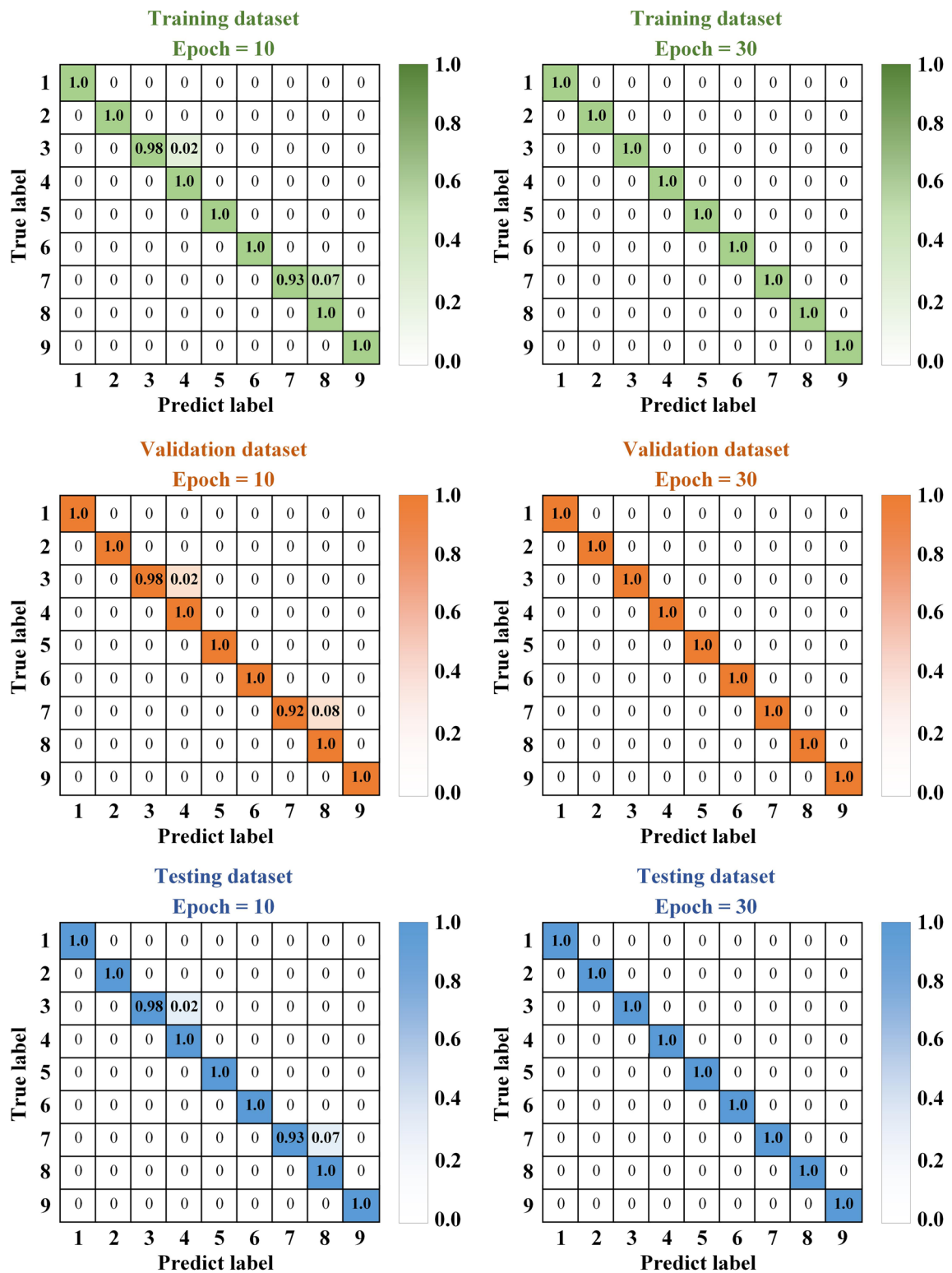
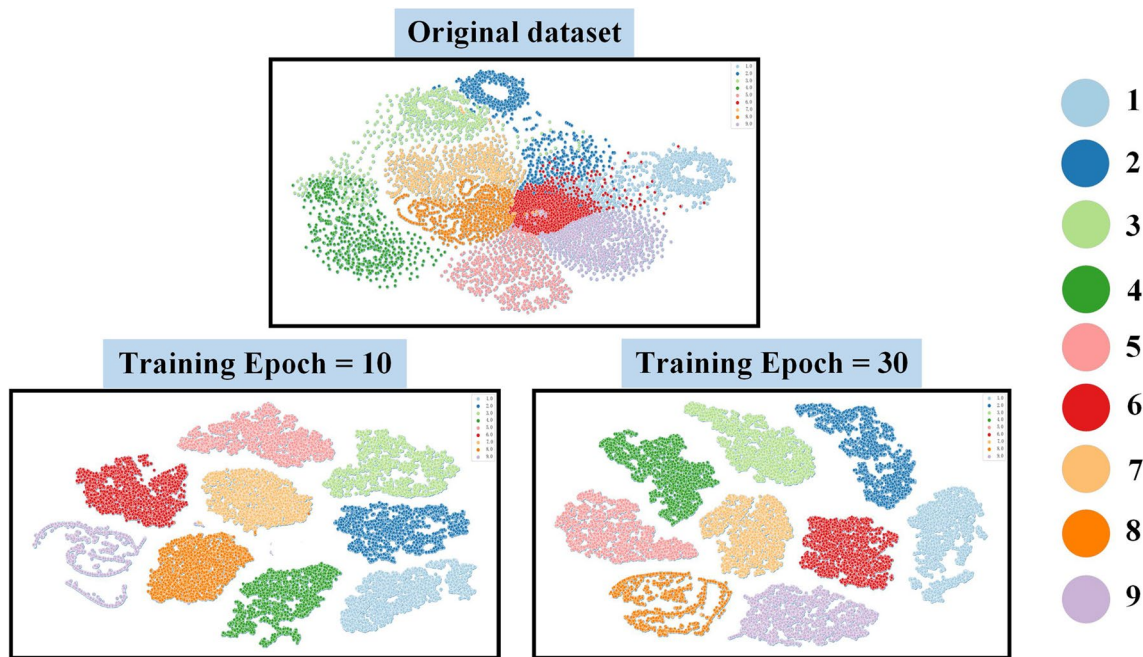**Fig. 16** Confusion matrix of damage type experiment

Fig. 17 T−SNE visualization

Table 3 Comparison results of model parameter quantity

Model	Parameter quantity
SGNet	1.0 million
MobileNet V1	4.2 million
ShuffleNet 0.5x	1.8 million
ShuffleNet 1.0x	2.3 million
GhostNet	5.2 million

- When the training epoch reaches 3, the accuracy of SGNet accuracy improves to 61.7%, whereas the accuracy of MobileNet reaches 18.8%, and the accuracy of the other two models is lower than 11%.
- When the training epoch reaches 20, the accuracy of SGNet is 97.9%, while the accuracy of the other three models is lower than 90%.
- Finally, with the training epoch reaches 30, the accuracy of SGNet achieves 99.8%.

In terms of average accuracy, SGNet's performance stands out, reaches 78.61%. This is notably higher than the average accuracies of MobileNet and ShuffleNet by 11.38% and 16.2%, respectively. The average accuracy of GhostNet is lower than 55%. Overall, SGNet consistently outperforms the other three models in terms of accuracy and exhibits faster convergence.

Comparison of the Accuracy of Other Methods

Ren et al. conducted experimental research by using the proposed BICCN [38] model to analyze data from 12 acceleration sensors under nine different damage cases. They obtained a set of samples suitable for structural damage localization and diagnosis for the frame structure. In order to ensure the reliability of the selected samples, experiments were conducted by using various models including 1DCNN, WDCNN [39], TICNN [40], ITICNN [41], and BICNN. The results are shown in Table 5; define results with an accuracy exceeding 95% as "excellent." Among these models,
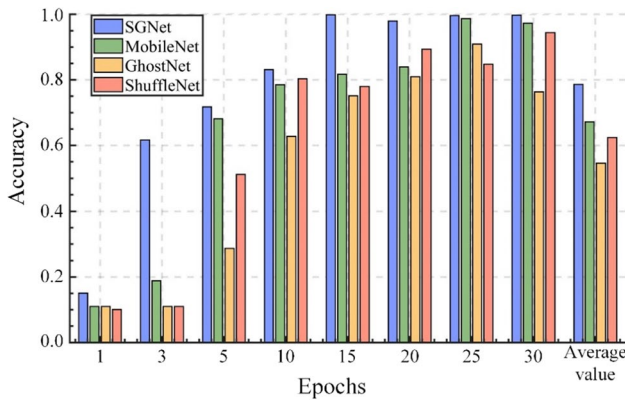
Table 4 Comparison results of testing dataset accuracy / (%)

Model	Epoch								
	1	3	5	10	15	20	25	30	Average Value
SGNet	15.1	61.7	71.8	83.2	99.8	97.9	99.6	99.8	78.61
MobileNet	10.9	18.8	68.1	78.5	81.7	83.9	98.6	97.3	67.23
GhostNet	10.9	10.9	28.7	62.7	75.2	80.9	90.9	76.3	54.61
ShuffleNet	10.1	10.9	51.1	80.4	78.0	89.4	84.8	94.4	62.41

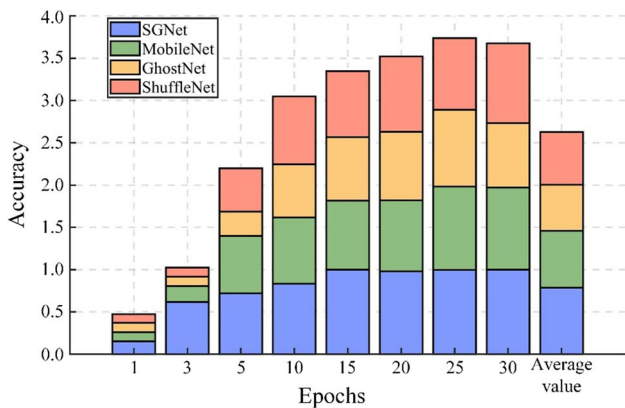**Fig. 18** Testing dataset accuracy (Bar chart)

Fig. 19 Testing dataset accuracy (Stack bar chart)

1DCNN had zero instances of excellence, while WDCNN, TICNN, ITICNN, and BICNN each had five instances of excellence. On the other hand, the SGNet model had 10

instances of excellence, demonstrating superior accuracy compared to the other models.

Based on the data from Table 5, it can be observed that the model's accuracy is relatively low at specific locations in the frame structure, specifically at locations numbered 5, 8, 11, and 14 in Fig. 7. This indicates that the model faces certain challenges or difficulties in damage diagnosis at the central pillars of the frame structure. This may be attributed to the fact that central pillars typically bear greater structural loads and stresses, making the detection and classification of damage more complex. Some types of damage may be harder for sensors to detect, leading to potentially higher levels of noise in the data from these locations, which affect the model's accuracy.

To improve the model's accuracy at these locations, it may be necessary to obtain more training data, especially focusing on damage cases related to central pillars. In addition, it is possible to consider improving the sensors layout to provide more reliable data. Overcoming these challenges will help improve the accuracy of damage diagnosis for the central pillars of frame structure.

Discussion

The SGNet model is an improved version based on ShuffleNet and GhostNe. It aims to reduce the computational cost of the model through a series of lightweight adjustments. These improvements include reducing the number of model parameters, decreasing network depth, and adopting more efficient network architecture. SGNet has achieves a better balance between performance and computational cost through carefully designed convolution kernels and step sizes, as well as parameter optimization, while maintaining high accuracy. Experimental results show that the highest accuracy of the model proposed in this article is 99.8%, with only 1 million parameters, and its performance is superior to other models.

Table 5 Comparison results of the accuracy of other methods

sensor	1DCNN	WDCNN	TICNN	ITICNN	BICNN	SGNet
4	40.36%	99.46%	99.88%	99.96%	99.99%	100%
5	55.38%	99.77%	99.73%	100%	100%	98.43%
6	49.21%	100%	100%	99.77%	99.99%	99.85%
7	50.11%	90.08%	90.34%	91.33%	93.98%	98.77%
8	21.22%	84%	76.92%	85.31%	90.01%	94.65%
9	42.75%	99.65%	99.96%	100%	100%	99.38%
10	26.21%	84.38%	77.70%	83.46%	89.63%	99.33%
11	29.15%	84.77%	78.35%	85.23%	88.74%	93.31%
12	32.35%	84.07%	78.04%	84.85%	90.23%	99.67%
13	35.75%	85.23%	79.05%	85.31%	90.38%	100.00%
14	47.43%	99.61%	99.81%	99.73%	100%	96.83%
15	59.03%	83.42%	78.16%	85.04%	87.56%	99.87%
Excellent	0	5	5	5	5	10

However, there are certain limitations to be noted. 1. This method may be sensitive to the quality and placement of sensors, thereby affecting the accuracy of diagnosis results. 2. It is limited to data from frame structures and may not be applicable to other structural types. 3. The model proposed in this article requires a large number of data samples for training, which may not be suitable for situations with fewer data samples. 4. Evaluation of the model was performed by using dataset of frame structures provided by Columbia University, but it may not cover all real-world application scenarios.

In future work, the authors plan to collect other publicly available structural datasets and apply transformations to the Columbia University datasets, including the introduction of noise with different signal-to-noise ratios to conduct research on damage diagnosis in noisy environments. In addition, the authors will explore alternative methods for structural damage diagnosis in cases with limited data samples, addressing the issue of lower accuracy when data is scarce.

Conclusion

In order to propose a more efficient damage diagnosis of frame structure while reducing computational costs, this paper introduced a lightweight model suitable for mobile devices, named SGNet. This model is an improvement on the ShuffleNet and GhostNet models. This model has stacked the ShuffleNet and GhostNet modules, and carefully designed the convolution kernel and step size for the first layer of the model. In addition, a data preprocessing method employing mean filtering had been successfully applied to the damage diagnosis of frame structure.

To evaluate the performance of the proposed method, a frame structure at Columbia University was selected as the experimental subject. Multiple damage diagnosis experiments were conducted by using the SGNet model, and the proposed model was compared with MobileNet, GhostNet, and ShuffleNet under the same conditions. The following conclusions were drawn from the experimental research.

(1) The experimental results of damage degree diagnosis indicated that the proposed SGNet model has the characteristics of fast convergence and high accuracy. In addition, the proposed SGNet model performed well in different damage cases and could determine the overall damage degree of the frame structure based on Pod_{avg}.

(2) The SGNet model exhibited strong multi-classification capabilities in the damage type diagnosis experiment. The accuracy of the testing dataset reached 99% when the training epoch reached 10. The proposed model could quickly diagnose damage types in damage cases of frame structures.

(3) In model comparison experiments, the SGNet model had the fewest parameters and the lowest computational cost compared to other models. Its accuracy on the testing dataset outperformed other models at different training epochs, and its average accuracy was the highest. The SGNet model also demonstrated the fastest convergence speed.

In summary, the experimental results of this paper unequivocally demonstrated the superiority of the SGNet model in the context of structural damage diagnosis for frame structure. It also highlighted the potential applicability of this model for mobile applications. Furthermore, the proposed preprocessing method provided valuable reference for similar tasks in research.

Declarations

Conflict of Interest The authors declare no conflict of interests.

References

1. Zeng Z (2020) Analysis and exploration of the application and development of neural network technology in mechanical engineering. Science and Technology Innovation and Application 18:153–154 (in Chinese)
2. Gul M, Catabas FN (2011) Structural health monitoring and damage assessment using a novel time series analysis methodology with sensor clustering. Sound of Sound and Vibration 33:1196–1210
3. Yin X (2013) Discussion on the existing problems and solutions in contemporary civil construction. Science and Technology Innovation and Application 32:241–248 (in Chinese)
4. Li H, Zhang Q, Qin X, Sun Y (2018) Bearing fault diagnosis method based on short−time Fourier transform and convolutional neural network. Vibration and Shock 37(19):124–131 (in Chinese)
5. Jiang J, Wang H, Ke Y (2017) Fault diagnosis based on LTSA and K−nearest neighbor classifier. Vibration and Shock 36(11):134–139 (in Chinese)
6. Cai E, Li C, Liu D (2015) Rolling bearing fault diagnosis method based on local mean decomposition and K−nearest neighbor algorithm. Modern Electronics Technique 38(13):50–52 (in Chinese)
7. Feng E, Pan H (2010) Research on diesel engine fault diagnosis based on fuzzy clustering analysis. Machinery Management and Development 1:25–26 (in Chinese)
8. Luo M, Zhang T, Meng X (2010) Fault diagnosis technology for aerospace launch based on fuzzy clustering analysis. Computer Engineering and Design 8:155–157 (in Chinese)
9. Su L, Guo J (2020) Research on lubricating oil particle detection based on improved ensemble empirical mode decomposition method. Journal of YanShan University 44(5):477–486 (in Chinese)

10. Li X, Law S (2010) Matrix of the covariance of covariance of acceleration responses for damage detection from ambient vibration measurements. Mech Syst Signal Process 24(4):945–956

11. Malekjafarian A, Brincker R, Ashory M (2012) Modified Ibrahim time domain method for identification of closely spaced modes: Experimental results. In: Proceedings of the 30th IMAC, A Conference on Structural Dynamics, Topics on the Dynamics of Civil Structures, Volume 1, 2012. Springer New York, pp 443–449

12. Yang J, Lei Y, Lin S (2004) Hilbert−Huang based approach for structural damage detection. J Eng Mech 130(1):85–95

13. Li H, Deng X, Dai H (2007) Structural damage detection using the combination method of EMD and wavelet analysis. Mech Syst Signal Process 21(1):298–306

14. Zhu X. Bearing fault diagnosis method based on wavelet packet decomposition and K−nearest neighbor algorithm. Equip Manuf Technol 2020(2), 24−27+45. (in Chinese)

15. Tong G (2020) Motor fault recognition method based on machine vision and experimental research. Experimental Technology and Management 37(4):71–77 (in Chinese)

16. Zhu H, Li Q, Ma T (2022) Application and fault analysis of machine vision in online wire harness assembly. Information and Computer 34(12):100–102 (in Chinese)

17. Krizhevsky A, Sutskever I, Hinton G (2012) Imagenet classification with deep convolutional neural networks. Adv Neural Inf Proces Syst:1097–1105

18. Devlin J, Chang M, Lee K (2018) BERT: Pre−training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805

19. Chen Y, Wang C (2012) Research on fault diagnosis method based on natural language understanding. Computer Measurement & Control 20(03):610–613 (in Chinese)

20. Li J (2022) Recent advances in end-to-end automatic speech recognition. APSIPA Transactions on Signal and Information Processing 11(1):1–64

21. Lee W, Seong J, Lu B (2021) Biosignal sensors and deep learning−based speech recognition: a review. Sensors 21(4):1399

22. Huang K, Shi B, Li X (2022) Multi−modal sensor fusion for auto driving perception: a survey. arXiv preprint arXiv:2202.02703

23. Bojarski M, Testa D, Dworakowski D (2016) End to end learning for self−driving cars. arXiv preprint arXiv:1604.07316

24. Kostić B, Gül M (2017) Vibration−based damage detection of bridges under varying temperature effects using time−series analysis and artificial neural networks. J Bridg Eng 22(10):04017065

25. Khodabandehlou H, Pekcan G, Fadali M (2019) Vibration−based structural condition assessment using convolution neural networks. Struct Control Health Monit 26(2):e2308

26. Avci O, Abdeljaber O, Kiranyaz S (2019) Convolutional neural networks for real−time and wireless damage detection. In: Proceedings of the 37th IMAC, A Conference and Exposition on Structural Dynamics, 2019, Volume 2. Springer International Publishing, pp. 129–136

27. Tang Z, Chen Z, Bao Y (2019) Convolutional neural network−based data anomaly detection method using multiple information for structural health monitoring. Struct Control Health Monit 26(1):e2296

28. Cuşkun Y, Kaplan K, Alparslan B. Classification of the brain metastases based on a new 3D deep learning architecture. Soft Comput 2023: 1–14

29. Al-Areqi F, Konyar MZ (2022) Effectiveness evaluation of different feature extraction methods for classification of covid-19 from computed tomography images: a high accuracy classification study. Biomedical Signal Processing and Control 76:103662

30. Yue S, Lei W, Xue Y (2021) Research on fault diagnosis method based on deep adversarial transfer learning. Mechanical Science and Technology:1–7 (in Chinese)

31. Zhang X, Zhou X, Lin M (2018) Shufflenet: an extremely efficient convolutional neural network for mobile devices. Proceedings of the IEEE conference on computer vision and pattern recognition. 6848–6856

32. Han K, Wang Y, Tian Q, Guo J (2020) GhostNet: more features from cheap operations. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition:1577–1586

33. Howard A, Zhu M, Chen B (2017) Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv:1704.04861, 1–9

34. He K, Zhang X, Ren S (2016) Deep residual learning for image recognition. Proc IEEE Conf Comput Vis Pattern Recognit:770–778

35. Krizhevsky A, Sutskever I, Hinton G (2012) Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems, In NeurIPS, pp. 1097–1105

36. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large−scale image recognition. arXiv preprint arXiv:1409.1556

37. Abdeljaber A, Avci O, Kiranyaz S (2018) 1−DCNNs for structural damage detection: verification on a structural health monitoring benchmark data. Neurocomputing 275:1308–17

38. Ren J, Cai C, Chi Y (2022) Integrated damage location diagnosis of frame structure based on convolutional neural network with inception module. Sensors 23(1):418

39. Zhang W, Peng G, Li C (2017) A new deep learning model for fault diagnosis with good anti-noise and domain adaptation ability on raw vibration signals. Sensors 17(2):425

40. Zhang W, Li C, Peng G (2018) A deep convolutional neural network with new training methods for bearing fault diagnosis under noisy environment and different working load. Mech Syst Signal Process 100:349–543

41. Xue Y, Cai C, Chi Y (2022) Frame structure fault diagnosis based on a high-precision convolution neural network. Sensors 22:9427