SCIENCE CHINA
Information Sciences

• **Supplementary File** •

Optical time-series signals classification based on data augmentation for small sample

Xuezhi Zhang[1,2,3], Haonan Sun[1,2,3], Junfeng Jiang[1,2,3*], Kun Liu[1,2,3], Zeyu Li[1,2,3], Jiahang Jin[1,2,3], Wenxin Bo[1,2,3], Yin Yu[1,2,3] & Tiegen Liu[1,2,3*]

[1]*School of Precision Instrument and Opto-Electronics Engineering, Tianjin University, Tianjin* 300072*, China;*
[2]*Tianjin Optical Fiber Sensing Engineering Center, Institute of Optical Fiber Sensing of Tianjin University, Tianjin* 300072*, China;*
[3]*Key Laboratory of Opto-Electronics Information Technology (Tianjin University), Ministry of Education, Tianjin* 300072*, China*

Appendix A Experiment setup

Appendix A.1 damage location detection experiments

As shown in Figure A1, the experimental system consists of two parts. One part is a platform for simulating material damage detection by ultrasonic signal. Standard ultrasonic signal is generated by a broadband acoustic emission PZT. As shown in Figure A2, an aluminum alloy plate with dimensions of 300 × 300 × 2 mm is used as the propagation carrier of the ultrasound signal. An active excitation point driven by PZT is placed in the right half of the horizontal centerline of the aluminum plate, 80mm away from the center. PZT is adhered to aluminum alloy plate with coupling agent in order to reduce the acoustic impedance difference between PZT and aluminum plate. The PZT is driven by a standard 500kHz sinusoidal signal with 0.02ms time duration. Before driving the PZT, the signal generated by function signal generator is amplified by a high voltage power amplifier. Ultrasonic longitudinal wave generated by PZT propagates through the aluminum plate. This ultrasonic longitudinal wave is picked up by the FBG placed in the center of the plate. Four positions on a circle from the center of the aluminum plate with 40 mm radius are marked as points to be detected, i.e., the positions 1, 3, 5 and 7 in Figure A2.

In the experiment, the damage of the material is simulated by placing 100g weight on the aluminum plate at the marking point. Three types of possible losses (11 Tags in total) are simulated here:

- Undamaged state: no weight is placed at any marked points (label: 0).
- Single point damage: a single weight is placed at a random point at the four marked points (labels: 1, 3, 5 and 7).
- Two-point damage: two weights are placed at the same time at two of the four marked points (labels: 1+3, 1+5, 1+7, 3+5, 3+7 and 5+7).

The other part is an FBG sensing system for monitoring the ultrasonic vibration, which is illustrated in Figure A1(b). The PS-FBG in this part has higher sensitivity than the normal FBG. Light source in the wavelength range of 1549.2 nm to 1550.5 nm is generated by a tunable fiber laser. The linewidth of the light source is up to 400kHz. The central wavelength of the laser output is adjusted in real time through the method of proportional-integral-derivative control strategy to ensure that it is always in the linear region of PS-FBG reflection peak [1]. A fiber optic isolator is connected after the laser to prevent the laser from damage induced by backward light in the optical path. By adjusting the gain of the fiber optic attenuator, the output intensity of the light source is controlled to avoid the photodetector reaching saturation. The light is split by a coupler. 5% of the optical power is used to monitor the power fluctuation of the light source, and the remaining 95% is used for ultrasonic signal sensing. The reflected light from PS-FBG is detected by a balanced photodetector (BPD), which can effectively improve the sensitivity of the optical sensing system while suppressing the noise [2].

Appendix A.2 Datasets

The experimental data was sampled by a data acquisition card at 4096 samples per frame. Two frames were saved together for each valid signal, i.e., a sequence of 8192 lengths. The results were filtered out by a digital bandpass filter to remove low frequency noise. In the experiment, a total of 2302 valid signals were collected and divided into a training set and a test set in the ratio of 8:2. The data amount of each class is shown in Table A1, and some examples of them are shown in Figure A3. As can be seen from the figures, the collected optical signals exhibit complex characteristics. The signal difference between different classes is so complex that it is difficult to classify by establishing an artificial model. Therefore, it is necessary to construct a deep learning model for classification. Normalization and data augmentation were performed on the collected data. The effect of data augmentation for a single sample is shown in Figure A4.

Appendix A.3 Model

In order to achieve the classification of one-dimensional signal samples, we consider two main 1D-CNN frameworks, as shown in Figure A5. Figure A5(a) shows a normal Plain 1D-CNN, where each CNN block consists of a convolutional layer, a batch normalization (BN) layer, an activation layer and a pooling layer. 1D-CNN with Residual Learning (RL) is shown in Figure A5(b). In this figure, network depth is deepened by residual connections in the normal convolutional layer. Two types of residual blocks are shown in Figure A6. One is a residual structure without downsampling, which extracts characteristics through two layers of

* Corresponding author (email: jiangjfjxu@tju.edu.cn, tgliu@tju.edu.cn)

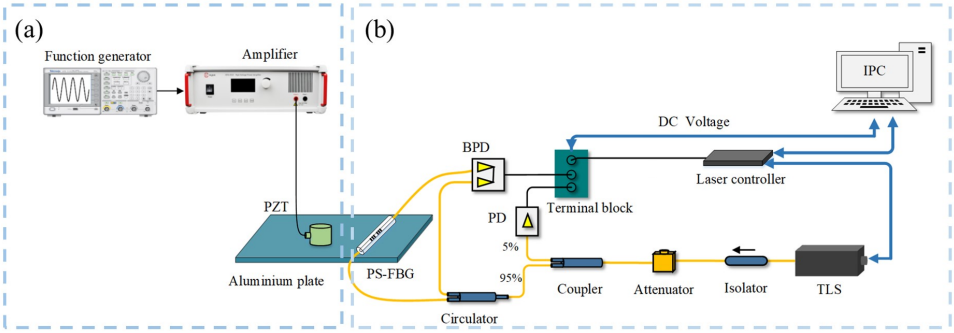

Figure A1 Schematic diagram of the system for damage location detection experiments. (a) Damage simulation platform for generating ultrasound signals using PZT. (b) PS-FBG ultrasound sensing system based on tunable narrowband laser demodulation technology.

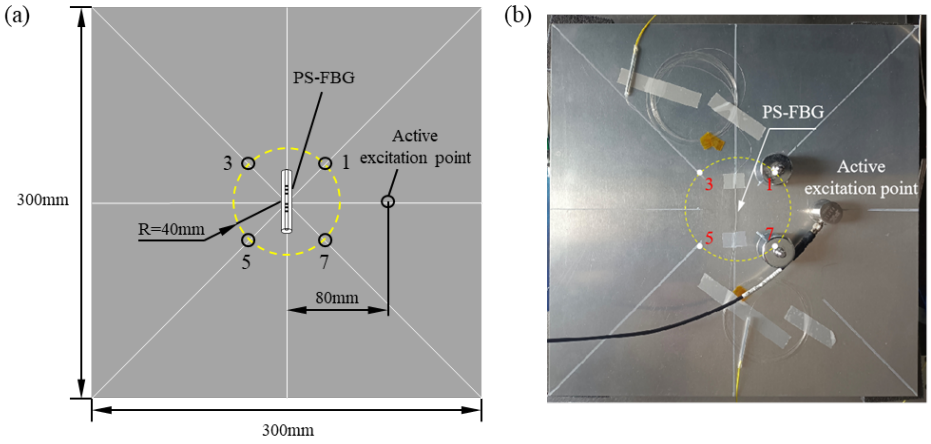

Figure A2 Monitoring point configuration for simulated damage platforms. (a) Schematic. (b) Actual configuration diagram.

Table A1 Number of data for different labels

Label	0	1	3	5	7	1+3	1+5	1+7	3+5	3+7	5+ 7
Total	200	229	165	207	238	165	205	287	166	223	217
Train set	159	188	126	168	200	129	162	223	126	182	178
Test set	41	41	39	39	38	36	43	64	49	41	39

convolution and passes them to deeper layers of the network. The other structure acts as a downsampling block besides the above functions.

Residual learning can solve the degradation problem of ordinary CNN. The original intention of introducing residual networks to solve this problem was to maintain at least the same performance of deep networks as shallow ones, i.e. to use residual layers to achieve constant transformation between block connections. The plain CNN have information loss between blocks due to the presence of a non-linear layer, as shown in Figure A7(a), making it difficult to achieve a direct fit of $H(x) = x$ with this layer. However, with the introduction of residual learning, the output $H(x) = F(x) + x$, as shown in Figure A7(b). To achieve a constant transformation, it is sufficient to let the $F(x) = 0$. (Whenever a feature less than 0 is learned by convolution, it becomes 0 after ReLU.) This is much easier than learning that $H(x) = x$. This ensures that in the worst case, even if the neural network does not learn anything, it can still achieve a constant transformation between layers, effectively avoiding adjustment towards a worse direction with the increase of the number of layers.

The L2 regularization method is also used in the loss design of the model to alleviate the overfitting problem. The regularization method is implemented by adding a penalty term to the objective function $J(\theta)$, which limits the learning capability of the model by compressing the parameter space. The L2 regularization strategy brings the parameter space of the weights closer to the origin. The modified objective function can be expressed as:

$$\tilde{J}(\theta) = J(\theta) + \alpha \|\theta\|_2^2 \tag{A1}$$

where α is the hyperparameter, which is set to 0.001 in the experiment.

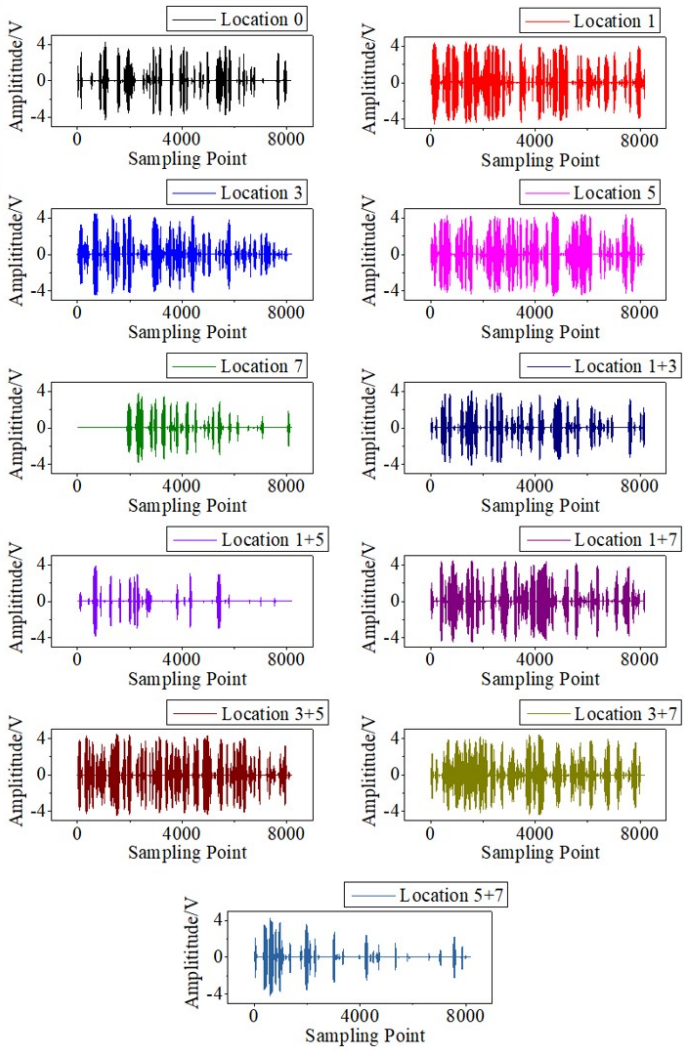

Figure A3 Results of partial data collection.

Appendix B Results

Appendix B.1 The choice of optimization algorithm

Optimization algorithms for the loss curve of a 1D-CNN with RL on the training set are selected based on the training error, as shown in Figure B1. The optimization algorithm with SGD train loss decreases relatively slowly, and the convergence speeds up significantly if momentum is introduced. Before 50 epochs, the convergence speed of Adam is the fastest. While the Adam curve gradually flattens out after 50 epochs. A train loss oscillation is even appeared in this area on Adam curve. Therefore, the SGD method with momentum for manual adjustment of the learning rate is finally chosen, which attenuates the learning rate to half after each certain epoch until convergence.

Appendix B.2 Results analysis

To analysis the results in detail, the impacts of different strategies on classification accuracy are compared firstly, as shown in Table B1. The experimental results show that for both Plain 1D-CNN and 1D-CNN with RL, data augmentation strategy alone can achieve an improvement in classification accuracy. When adopting only this strategy, random scale-cropping provides a greater gain because it expands the data amount significantly. Random erasing, on the other hand, selects out important characteristics in terms of statistical properties, thus suppressing overfitting of the model to specific characteristics. However, this improvement is weak in the case of inadequate data sets. When both strategies are adopted at the same time, random erasing exhibits good statistical properties and is more helpful to select important characteristics due to the suppressing of overfitting problem. As can be seen from Table B1 after data augmentation, both models obtained a huge improvement in accuracy on the test set compared to the result obtained when no strategy is taken.

The classification ability of normal MLP and CNN has been compared by Top-K error. As shown in Table B2 and Table B3, Top-K error represents the error rate on the top k classification results with the highest probability of being selected. The classification result is considered correct if one of the k results contains the correct label, and incorrect if the opposite is true. When analysis the result without data augmentation strategy, all three models fail to perform satisfactorily in terms of classification accuracy. With the data augmentation strategy, the classification performance of the CNN is significantly improved, and the Top-3 error is below 4% for both Plain CNN and 1D-CNN with RL. For the MLP, data augmentation fails because the MLP calculates the output

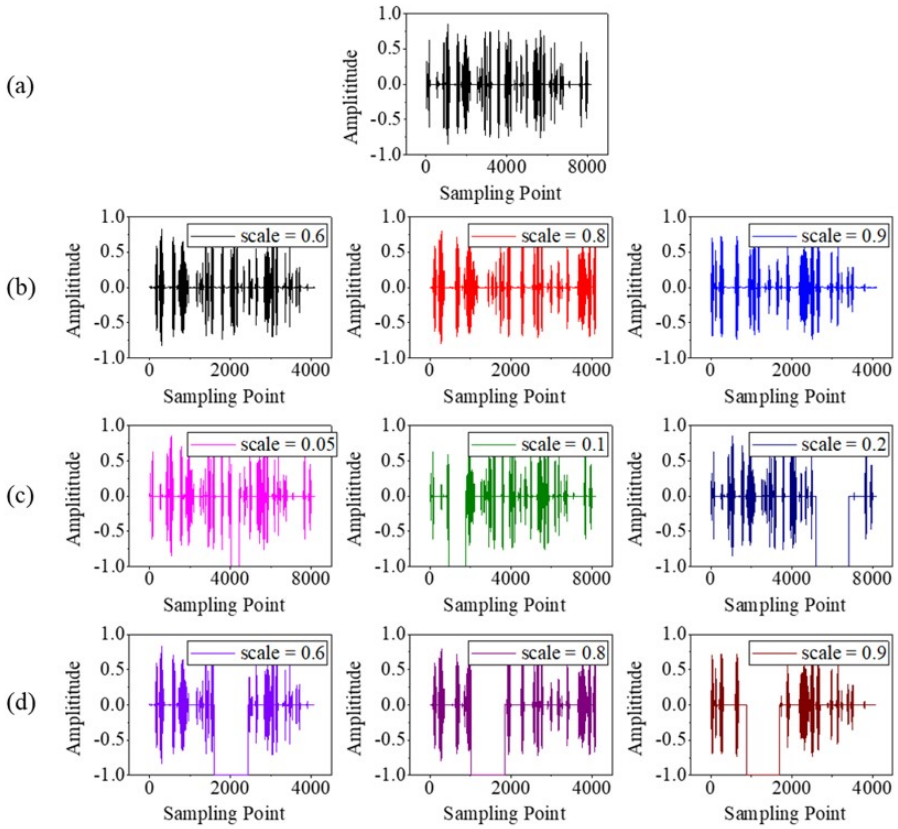

Figure A4 Data augmentation is applied to a single sample. (a) Original signal. (b) Data obtained using random scale-cropping with scales of 0.6, 0.8 and 0.9. (c) Data obtained by random erasing with a scale of 0.05, 0.1 and 0.2, respectively, where the value after erasing is set to -1. (d) Combination of two strategies, erasing scale set to 0.2, varying the data obtained with different scale-cropping scales.

values in a strictly fully-connected mode, which is specific and strict requirements for the value at each location. The extraction of characteristics by CNN, on the other hand, exhibits a statistical property in the local domain and is therefore more robust to the dataset when data augmentation is taken. This further demonstrates that CNN has a significant advantage over MLPs when dealing with time-series sequence. While in the case of the CNN without data augmentation, the 1D-CNN with RL doesn't show better classification ability than the Plain 1D-CNN, even with higher Top-3 error values. This is because the complex model is more likely to fall into overfitting when the data amount is small. After data augmentation, the classification performance of 1D-CNN with RL is better than Plain 1D-CNN.

We measured the distribution of data for each category after training by different routes, as shown in the Figure B2. The original distribution has a Gaussian distribution over the whole space and it is not possible to distinguish between the categories. With the model trained without data augmentation, individual categories show significant spikes in space, but most of the data remains stacked in the center. This indicates that the model is overfitting to some specific categories. After the model has been trained with the data augmentation process, it can be seen that each category shows a unique distribution in space. The columnar accumulation also show that the data is not stacked together and therefore the individual categories can be easily identified.

Figure B3 shows distribution of test set samples in space under different data augmentation methods. This implies how data augmentation methods work in the data space. A good model should ensure that there are clear boundaries between different classes of data, but avoid overlearning some features. Data augmentation is the process of inputting various artificial 'difficulties' so that the model does not learn some features so easily and avoids overfitting the model to specific classes. Figure B3(b) shows the distribution without any data augmentation of the output features. The data shows a high variance sample with data points far from the data center, and thus a more severe overfitting can occur. The spatial distribution of the data in Figure B3(c) and Figure B3(d) shows that both data augmentation techniques 'pull back' the distribution of the data in space towards the center, reducing the variance of the data and thus alleviating overfitting. And what is also evident from the figure is that the use of random scale-cropping is more effective than random erasing in reducing the variance of the data. The smaller the variance of the distribution of the data in space, the less likely overfitting will occur.

Table B4 shows the classification results of the 1D-CNN with RL model on each class using data augmentation in detail. From these results, the methods proposed in this paper show superior classification ability for both single-point damage and two-point damages, with an overall classification accuracy of 90.46% on the test set. Therefore, it is demonstrated that the model exhibits excellent classification performance.

Appendix B.3 A comparison of some other methods with our methods.

Table B5 lists some other common data augmentation. The table shows the advantages of our method in dealing with complex one-dimensional signals. Some of these means are not obvious enough, and temporal flipping is worse because our samples are not periodic.

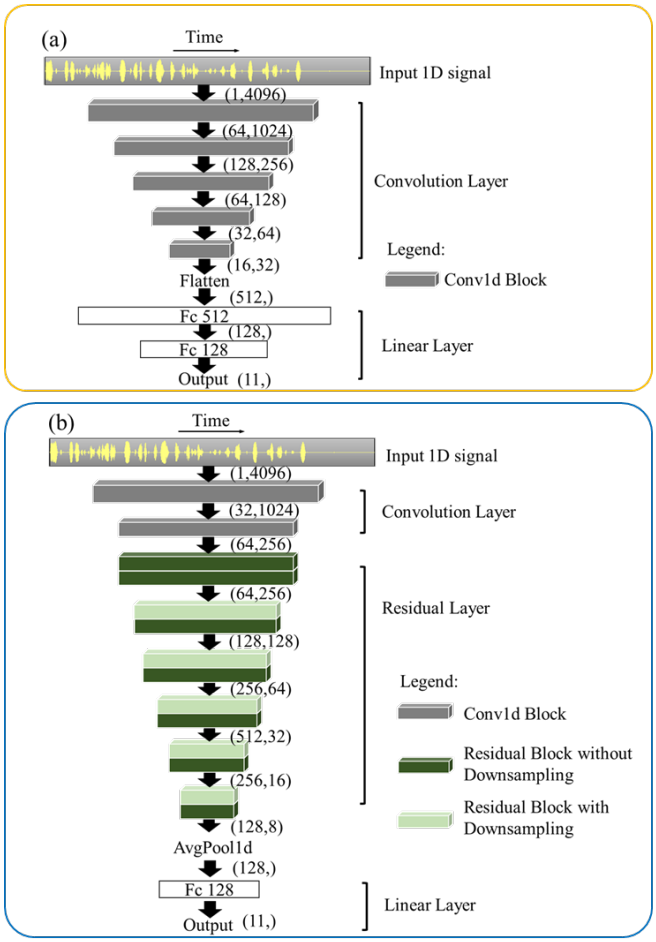

Figure A5 Schematic framework of the 1D-CNN damage recognition algorithm model. (a) Plain 1D-CNN. (b) 1D-CNN with RL.

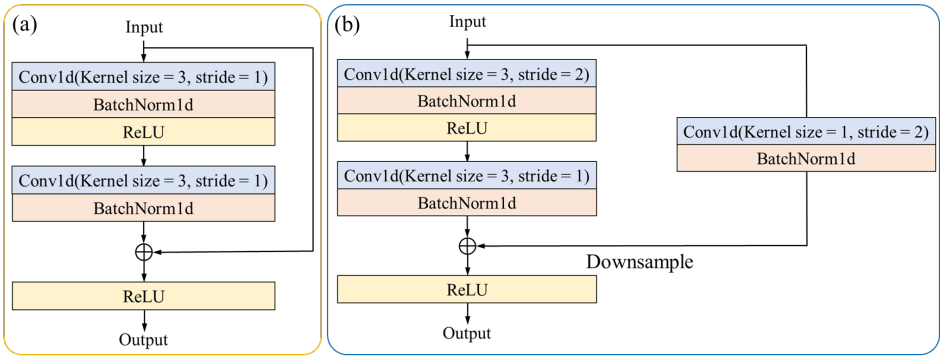

Figure A6 Schematic diagram of the configuration of the residual block parameters. (a) Residual block without downsampling. (b) Residual block with downsampling.

Appendix B.4 More situations

We have also measured the same three points where damage occurred. For the new case, a total of 512 new data sets were collected and divided into a training set and a test set according to 8:2. The system also showed good classification performance. Its confusion matrix is shown in the table B6. The final accuracy is 90.64%.

We have also observed the spatial distribution of the data, as shown in the Figure B4. The two main characteristic components of the data were plotted as a two-dimensional scatter plot. Due to the large number of damage cases, only a small portion of the data was selected for the remaining cases except for the three points and the no damage condition.

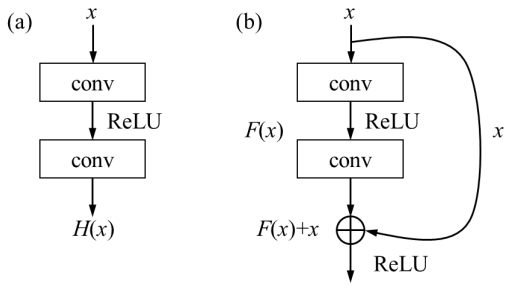

Figure A7 (a)Plain CNN block. (b)CNN with residual block.

Table B1 Effect of data argument on classification accuracy

Method	Strategy	Acc(test set)
Plain 1D-CNN	No strategy	52.28%
	Random erasing only	53.36%
	Random scale-cropping only	78.09%
	Both	85.47%
1D-CNN with RL	No strategy	51.19%
	Random erasing only	57.05%
	Random scale-cropping only	77.00%
	Both	90.46%

Table B2 Error rates of different methods on test set without data augmentation

Method	Top-1 error	Top-3 error	Top-5 error
MLP	65.29%	50.80%	35.14%
Plain 1D-CNN	47.72%	23.21%	11.71%
1D-CNN with RL	48.84%	30.59%	18.22%

Table B3 Error rates of different methods on test set with data augmentation

Method	Top-1 error	Top-3 error	Top-5 error
MLP		Non-convergence	
Plain 1D-CNN	14.53%	3.91%	2.93%
1D-CNN with RL	9.54%	3.47%	1.52%

Table B4 Confusion matrix of the classification results on test set

Confusion matrix		Classification results										
		0	1	3	5	7	1+3	1+5	1+7	3+5	3+7	5+7
Label	0	37	0	0	0	1	1	0	0	0	2	0
	1	0	41	0	0	0	0	0	0	0	0	0
	3	0	2	36	0	0	0	0	0	0	0	1
	5	0	0	1	36	1	0	0	0	0	1	0
	7	0	0	0	1	37	0	0	0	0	0	0
	1+3	0	0	0	0	0	36	0	0	0	0	0
	1+5	0	0	0	0	0	1	38	3	0	0	1
	1+7	1	0	0	0	2	0	1	54	0	0	6
	3+5	0	1	1	5	1	0	0	1	31	0	0
	3+7	0	0	0	1	1	0	0	1	0	38	0
	5+7	0	2	0	0	0	0	1	2	0	0	33

References

1 Liu Z Z, Zhang X Z, Jiang J F, et al. Stabilization of high sensitivity optical fiber AE sensing for long-term detection. Optical Fiber Technology, 2021, 61: 102391
2 Wu Q, Yu F M, Okabe Y, et al. Application of a novel optical fiber sensor to detection of acoustic emissions by various damages in CFRP laminates. Smart Mater Struct, 2015, 24: 015011

Table B5 Accuracy of different methods on test set

Method	Accuracy
No strategy	51.19%
Temporal flipping	18.02%
Amplitude flipping	53.00%
Gaussian noise	53.12%
Value shift	82.46%
Our methods	90.46%

Table B6 Confusion matrix of the classification results on test set

Confusion matrix		Classification results														
		0	1	3	5	7	1+3	1+5	1+7	3+5	3+7	5+7	1+3+5	1+3+7	1+5+7	3+5+7
Label	0	40	0	1	1	0	0	0	0	1	0	1	0	0	1	1
	1	0	41	1	1	0	0	0	1	0	0	0	1	0	0	0
	3	0	0	36	1	1	0	1	0	1	0	0	0	1	0	0
	5	0	0	0	33	1	0	0	0	2	2	1	0	0	1	0
	7	0	0	0	1	35	0	0	1	1	1	0	0	0	1	0
	1+3	0	0	0	0	1	35	0	0	0	0	0	0	0	0	0
	1+5	0	0	0	0	0	0	41	1	0	0	0	0	0	0	0
	1+7	0	0	0	0	0	0	1	52	1	0	0	0	0	1	0
	3+5	0	0	0	0	0	0	0	1	33	0	0	0	0	0	0
	3+7	1	0	0	2	0	0	0	1	0	38	0	0	0	0	0
	5+7	0	0	0	0	0	0	0	4	0	0	37	0	0	0	0
	1+3+5	0	0	0	0	0	0	0	1	0	0	0	22	0	0	0
	1+3+7	0	0	0	0	0	0	0	1	0	0	0	0	24	0	1
	1+5+7	0	0	0	0	0	0	0	1	1	0	0	1	0	18	0
	3+5+7	0	0	1	1	0	1	0	0	0	0	0	0	1	0	19

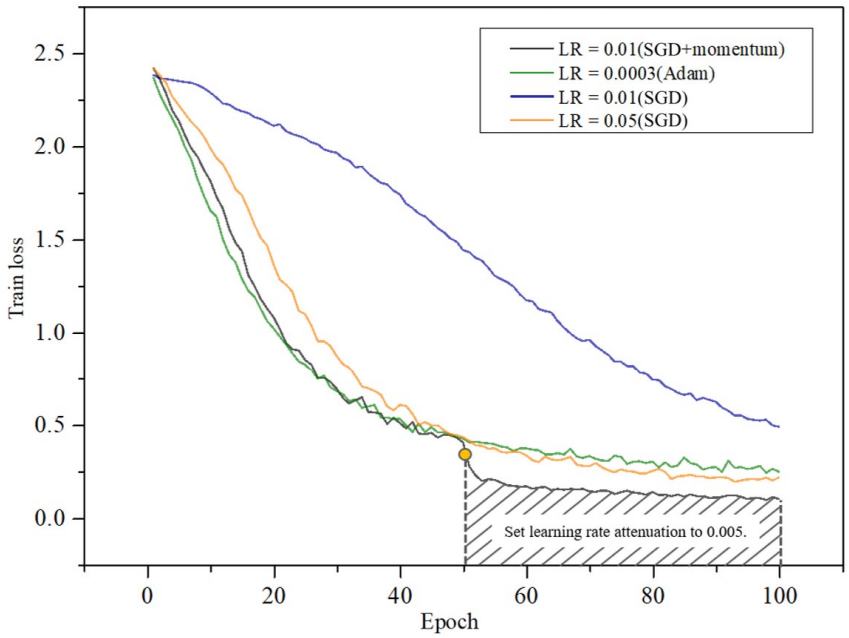

Figure B1 Performance on a 1D-CNN with RL model, comparison of training loss descent curves using different optimization algorithms.

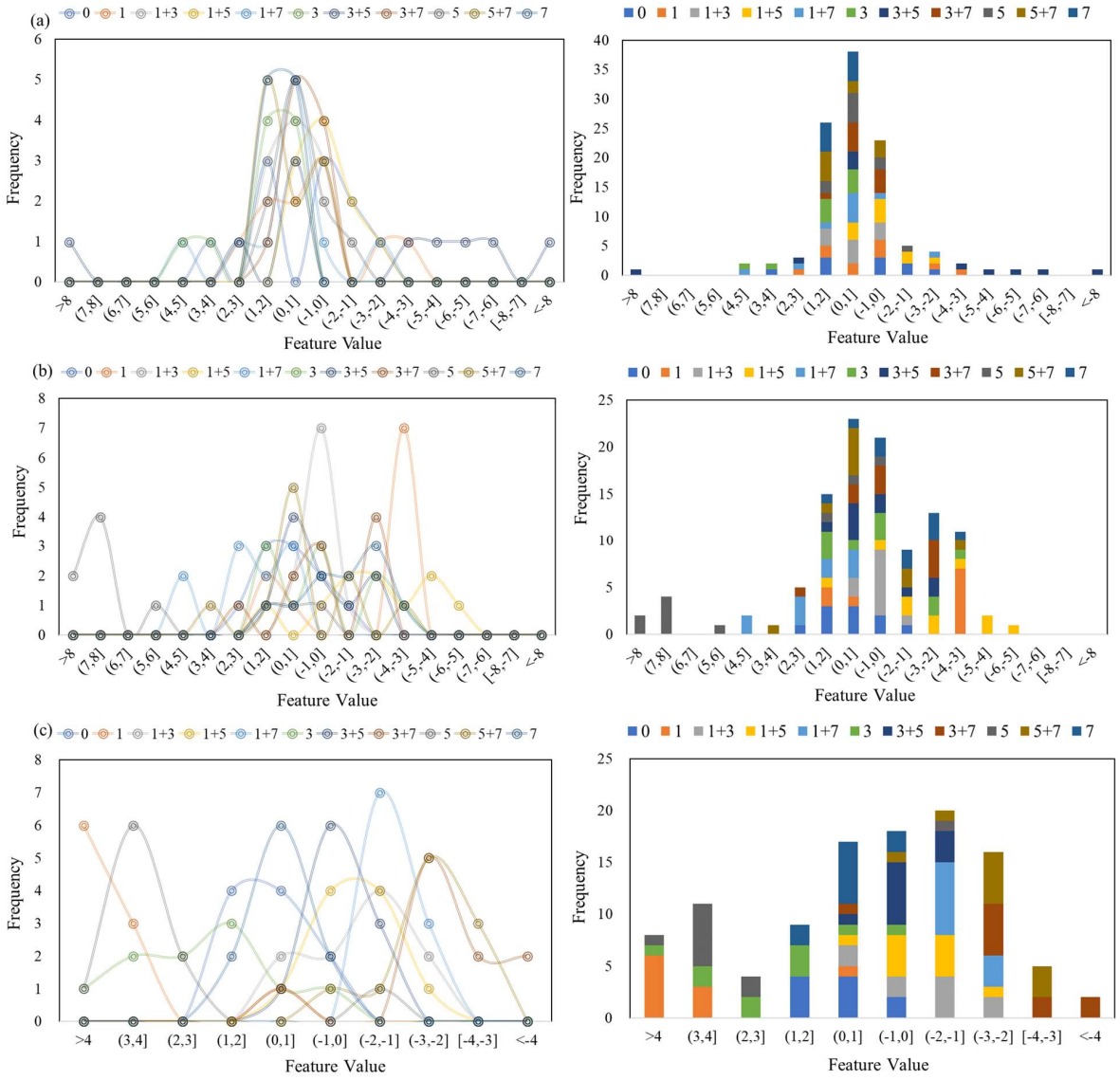

Figure B2 Schematic diagram of the distribution curves and columnar accumulation of each category in the main feature direction of the test set samples. (a) Original distribution. (b) Distribution of the test set after training the model without data augmentation. (c) Distribution of the test set after training the model without data augmentation.

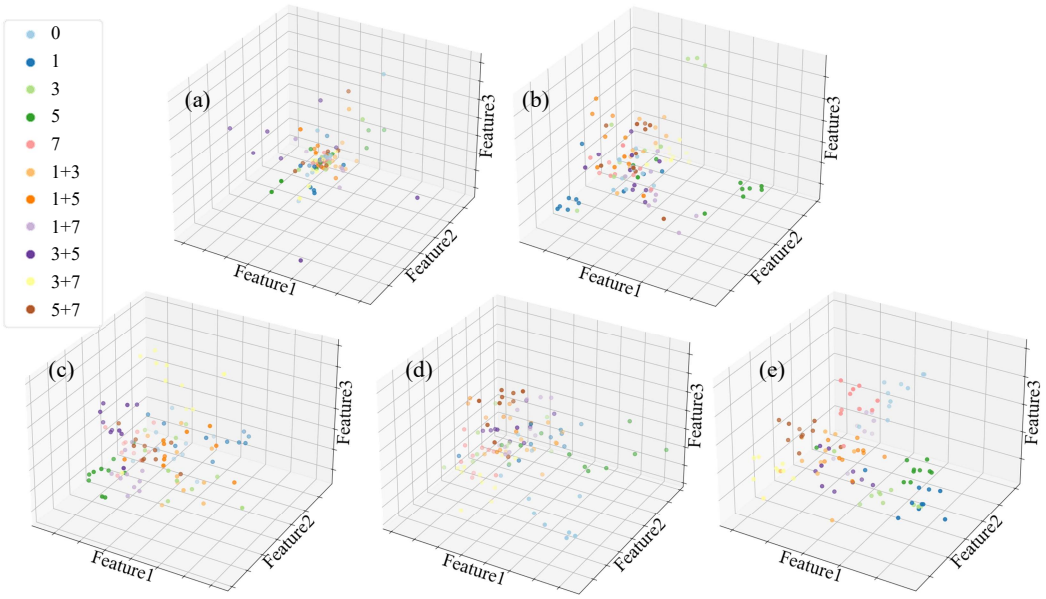

Figure B3 Schematic representation of the distribution of data features in space. (a) Original distribution. (b) No strategy. (c) Random scale-cropping. (d) Random erasing. (e) Both.

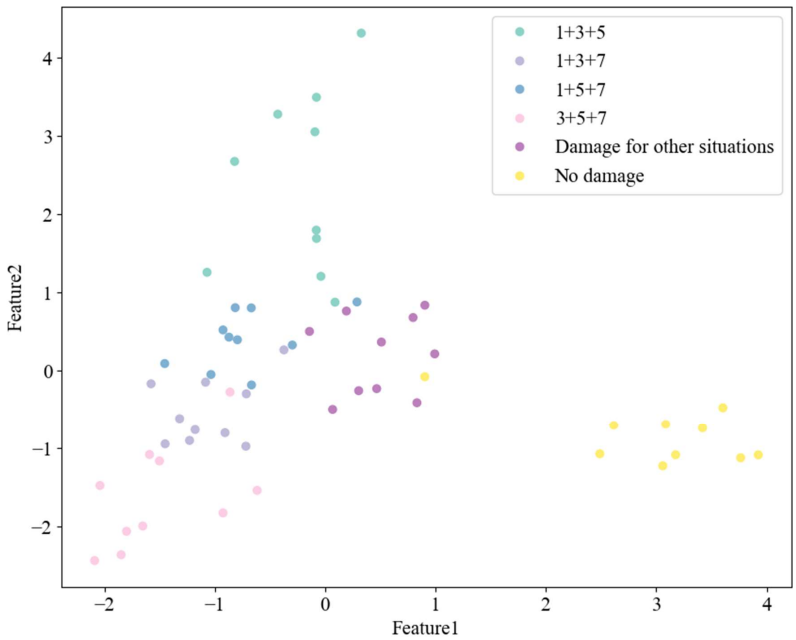

Figure B4 Two-dimensional scatter plot of data distribution for more situations.