Proceeding of the IEEE
International Conference on Robotics and Biomimetics
Dali, China, December 2019

CWGAN: Conditional Wasserstein Generative Adversarial Nets for Fault Data Generation

Ying Yu , Bingying Tang, Ronglai Lin, Shufa Han, Tang Tang, Ming Chen
School of Mechanical Engineering
Tongji University
Shanghai, China
1830204@tongji.edu.cn

***Abstract* - With the rapid development of modern industry and artificial intelligence technology, fault diagnosis technology has become more automated and intelligent. The deep learning based fault diagnosis model has achieved significant advantages over the traditional fault diagnosis method. However, a problem has arisen when deep learning models are applied to actual industrial scenarios. In the actual process of industrial production, there are not enough fault data for deep learning, hence the accuracy will decrease because of overfitting. In this paper, aiming at the problem, the fault data generation based on deep learning is deeply studied.**

In this paper, the data source is the experimental data from the Fault Data Center of Case Western Reserve University.

Aiming at the problem of small amount of fault data, a method of generating fault time-frequency spectrum based on improved conditional generative adversarial networks is proposed. CWGAN (Conditional Wasserstein Generative Adversarial Nets) learns the feature of time-frequency spectrum of rolling bearing fault, and generates time-frequency spectrum of corresponding fault categories according to the input categories. Experiments show that the diversity and fidelity of data generated by CWGAN is better than that of the original generative adversarial networks. The VGG-Net model is used to train the fault data enhanced by CWGAN. It is found that the data generated by CWGAN can effectively supplement the small amount of fault data, improve the training effect of the model and avoid over-fitting.

Index Terms* - *Rolling bearing, fault diagnosis, generative adversarial networks, deep learning, CWGAN

I. INTRODUCTION

The fault diagnosis is based on the operating state of the equipment. It identifies whether the device is faulty through data processing and analysis, and further determines the location of the fault, and finally guides the maintenance and repair of the equipment. The traditional fault diagnosis method is generally based on expert experience and signal processing technology. What's more, the original signal is designed by hand and the simple classification algorithm is used for fault identification[1][2].

In this case, due to the manual design features, a better diagnostic effect can be achieved only relying on strong engineering experience. So it is difficult to dig deeper into the potential information of the data.

Drawing on the successful experience of deep learning in the fields of natural language processing and image recognition[3-6], deep learning solves the problem of deep mining data information in the field of fault diagnosis [7-11].With the deep development of deep learning, deep learning networks have been proposed by researchers to solve the corresponding tasks, such as auto-encoder[12], the convolution neural network(CNN)[13], probabilistic based deep belief network[14] (DBN) , recurrent neural network (RNN) [15], and generative adversarial networks (GAN)[16].

Most of the current research is still limited to the identification of fault data collected by the laboratory, while in the environment with a small amount of data, the generalization performance of the model is still flawed. So applying the previous deep learning method directly to industrial data with small data volume is not ideal. There are two ways to solve the problem of small amount of data. The first way is to obtain new data, which requires a lot of cost, and the second one is data enhancement. Using data enhancement can increase the amount of data and improve the generalization ability of the model. Data enhancement can be divided into two categories[17] , offline enhancement[18] and online enhancement. Data enhancement methods include rotation, translation, scaling, and more. This paper is inspired by GAN, using GAN to generate fault data to achieve the purpose of supplementing fault data. GAN is a deep learning model that combines a generated model with a discriminant model. Through the confrontation training between the generator and the discriminator, the GAN can effectively learn the data features and generate data close to the training data. In the field of image generation and style migration, GAN has a large number of applications. In 2017, Casey Chu proposed CycleGAN[19] , which realized two image styles transformation by introducing loop consistency requirements. Reference [20], CycleGAN is applied to data enhancement. In the field of fault diagnosis, reference [21] builds CDCGAN, and adds the control tag to the input noise vector to realize the sample generation of the compound fault time-frequency diagram, but there are still problems of poor diversity of generated data and low quality of the generated time-frequency map. GAN's research in the field of fault diagnosis is still in its preliminary stage. So it has a good prospect to supplement GAN for fault data. This paper builds GAN to learn the characteristics of fault time-frequency diagram, and generates a lot of realistic fault time-frequency maps, which complements the fault data volume.

II. PROPOSED CWGAN FOR FAULT DIAGNOSIS

Reference [22] pointed out that the conventional signal processing method can mine the periodic component and the signal variation law, which is helpful to improve the extraction

978-1-7281-6321-5/19//$31.00 ©2019 IEEE

ability of the model feature. Therefore, this paper firstly processes the fault data, and then proposes a deep learning method to supplement the fault data.

Firstly, the rolling bearing vibration signal is preprocessed to generate an identifiable fault spectrum map to highlight the fault characteristics in the vibration signal.

Then introduce a generative adversarial network algorithm, which solves the problem of insufficient amount of fault data by generating a fault time-frequency map.

A. Rolling bearing fault diagnosis data preprocessing

This paper used wavelet transform for data preprocessing. Compared with the two commonly used methods of data transformation analysis, Fourier transform and short-time Fourier transform[23], wavelet transform not only has the characteristics of short-time Fourier transform local processing data, but also avoids the shortcoming of short-time Fourier transform window function. Therefore, wavelet transform can do better time-frequency analysis of non-stationary signals.

Based on the diversity of wavelet basis functions, we analyze the wavelet basis function and then select the optimal wavelet basis function to extract the most significant feature of the time domain signal during the processing of the rolling bearing fault signal. At the same time, the scale parameters need to be selected accordingly. Choosing the appropriate scale parameters can effectively denoise the signal to improve the subsequent identifiability of the wavelet time-frequency diagram.

Firstly, the characteristics of the rolling bearing fault signal are analyzed. In the case of a local failure of the rolling bearing, the time-series vibration signal collected by the acceleration sensor has an impact component of a part of the fault characteristic frequency. Therefore, how to extract the impact components with the corresponding frequency as the cycle is the key to selecting the wavelet basis function. In the actual industrial signal, the rolling bearing signal composition is complex, and the industrial original vibration timing signal has the characteristics of low signal-to-noise ratio, large noise disturbance and complex signal components, and the signal cannot be directly processed by the low-pass filter. Rolling bearing components include inner ring, outer ring and rolling element. In the event of bearing failure, the surface of the component is generally scratched, pitting, broken, etc. These damages will cause the chronological vibration signal to oscillate [24]. In order to identify the fault from the complex and low signal-to-noise ratio rolling bearing vibration signal, it is necessary to select the appropriate wavelet basis function to clearly represent the characteristics of the fault signal attenuation oscillation to highlight the characteristics of the fault. The selected wavelet basis function should approximate the waveform of the fault signal attenuation oscillation. Morlet wavelet basis function $\Psi(x) = Ce^{-\frac{x^2}{2}}\cos(5x)$ is a cosine signal with a square exponential decay.

B. CWGAN principle and algorithm flow

In the industrial field, the lack of fault data severely reduces the accuracy of the diagnostic method. With the development of GAN, by learning the characteristics of the fault samples and generating the fault data close to the real sample, the fault samples lacking in the data can be supplemented, thereby improving the recognition accuracy of the data-driven fault diagnosis method. GAN was proposed by Lan Good-fellow in 2014 as a deep learning model. GAN is a type of neural network used in unsupervised learning, mainly composed of a generator and a discriminator.

The optimization goal of GAN model is a maximal and minimal game problem, as shown in (1):

$$\min_G \max_D V(D,G) = E_{x\sim p_{data}(x)}[logD(x)] + E_{x\sim p_z(z)}\left[log\ (1 - D(G(z)))\right] \tag{1}$$

Following GAN, WGAN was published. The WGAN network improves the original GAN network optimization goal[25] based on the Wasserstein distance. The Wasserstein distance is defined as shown in (2):

$$\mathrm{W}(P_r, P_g) = \inf_{\gamma\sim\Pi(P_r,P_g)} E_{(x,y)\sim\gamma}[\|x-y\|] \tag{2}$$

Where $\Pi(P_r, P_g)$ is the set of all possible joint distributions in which P_r and P_g are combined. For each possible joint distribution γ, a real sample x and a generated sample y can be sampled, and the sample distance $\|x-y\|$is calculated, so that the expected value $\gamma\, E_{(x,y)\sim\gamma}[\|x-y\|]$ of the sample to the distance under the joint distribution γ can be calculated. This expected value can be taken to the lower bound in all possible joint distributions and defined as the Wasserstein distance of the two distributions. The Wasserstein distance can be used to provide meaningful gradients for gradient descent training.

But WGAN does not really address the Lipschitz constraint.

In order to solve the problem of WGAN, a WGAN network based on the gradient penalty method(WGAN-GP) is proposed. By constraining the network output relative to the gradient norm of its input, a Gradient Penalty (GP) [26] is added to the optimization function, and the norm of the gradient is limited to a value of 1 to satisfy the 1-Lipschitz continuous condition. The WGAN-GP network improved the optimization function to (3):

$$\mathrm{L} = \mathop{\mathrm{E}}_{\tilde{x}\sim P_g}[D(\tilde{x})] - \mathop{\mathrm{E}}_{x\sim P_r}[D(x)] + \lambda\cdot GP$$
$$GP = \mathop{\mathrm{E}}_{\hat{x}\sim P_{\hat{x}}}[(\|\nabla_{\hat{x}}D(\hat{x})\|_2 - 1)^2] \tag{3}$$

Where GP is the gradient penalty term. $\hat{x}$ is obtained by sampling from the sample space of the $P_{\hat{x}}$-distribution. The $P_{\hat{x}}$-distribution is defined as a sample space formed by random sampling between a pair of points sampled in the real data distribution P_r and the generated data distribution P_g.

CGAN is an extension to the original GAN[27] .Both the generator and the discriminator add data category information y. The optimization function of CGAN is a maximal and minimal game with condition y.

Based on the idea of WGAN-GP, this paper uses WGAN-GP optimization method to optimize the CGAN. The conditional generation adversarial network based on the WGAN-GP idea designed in this paper is called CWGAN, and the loss function of CWGAN is modified to (4).

$$Loss_{WGAN-GP} = \mathop{\mathrm{E}}_{z\sim p_z(z)}\left[D\left(G(z|y)\right)\right] - \mathop{\mathrm{E}}_{x\sim P_r}[D(x|y)] + \lambda\cdot \mathop{\mathrm{E}}_{\hat{x}\sim P_{\hat{x}}}[(\|\nabla_{\hat{x}}D(\hat{x}|y)\|_2 - 1)^2] \tag{4}$$

The algorithm training process of CWGAN is shown in Algorithm 1:

Algorithm 1 CWGAN algorithm training process

CWGAN algorithm: η, the learning rate. m, the batch size. α, a random sample in interval [0, 1].

While θ has not converged **do**

- Sampling m noise samples$\{z^{(1)}, \ldots, z^{(m)}\}$ as a small batch sample from prior distribution $p_z(z)$,
- Sampling m real samples and corresponding labels, $\{(c^{(1)}, x^{(1)}), \ldots, (c^{(m)}, x^{(m)})\}$ as a small batch sample from real data distribution $p_{data}(x)$,;
- Get the generated data through the generator G:$\{\tilde{x}^{(1)}, \ldots, \tilde{x}^{(m)}\}$, where $\tilde{x}^{(i)} = G(c^{(i)}, z^{(i)})$
- Gradient descent method to update discriminator parametersθ_d, minimizing the loss function L, where$\hat{x}^i$ is a linear sample between the generated data $\tilde{x}^i$ and the real data x^i:

$$L \leftarrow \frac{1}{m}\sum_{i=1}^{m} D(c^i, \tilde{x}^i) - \frac{1}{m}\sum_{i=1}^{m} D(c^i, x^i) + \lambda \cdot GP$$

$$\hat{x}^i \leftarrow \alpha \cdot \tilde{x}^i + (1-\alpha) \cdot x^i$$

$$GP \leftarrow \sqrt{\frac{1}{m}\sum_{i=1}^{m} (\|\nabla_{\hat{x}} D(\hat{x}^i|y)\|_2 - 1)^2}$$

$$\theta_d \leftarrow \theta_d + \eta \nabla L(\theta_d)$$

- Sampling m noise samples$\{z^{(1)}, \ldots, z^{(m)}\}$ as a small batch sample from prior distribution$p_z(z)$,;
- Sampling m conditions (labels) from the data set$\{c^{(1)}, \ldots, c^{(m)}\}$
- Gradient descent method to update discriminator parametersθ_d, and optimize L:

$$L \leftarrow \frac{1}{m}\sum_{i=1}^{m} D\left(G(c^i, z^i)\right)$$

$$\theta_d \leftarrow \theta_d + \eta \nabla L(\theta_d)$$

end while

C. *CWGAN model structure design*

The model structure design of CGAN is based on the idea of DCGAN[28], and the convolutional neural network is applied to the construction of generator and discriminator.

In the construction of neural network, considering the complexity of the internal data of the wavelet time-frequency diagram, in order to increase the fitting ability of the model, some fully connected layers are retained. Based on the idea of CGAN, the input and the middle layer of the model adopt the method of splicing the label vector. At the same time, with the idea of DCGAN, the batch normalization (BN layer) is introduced to deal with the training problem caused by poor initialization, which also helps the gradient to flow to a deeper network and prevent the training generator from crashing.

In the network inference phase, since one sample is injected into the network at a time, the mean and variance in the batch cannot be calculated, so a moving average is used to maintain a record of the mean and the variance. At each update, (5) is used as the mean and variance of the current training step:

$$\begin{aligned} \mu_t &= 0.1 * \mu_t + 0.9 \times \mu_{t-1} \\ \sigma_t &= 0.1 * \sigma_t + 0.9 \times \sigma_{t-1} \end{aligned} \tag{5}$$

When actually building the model of this paper, the deconvolution is used to perform the upsampling of the feature in the generator. In this paper, micro-step convolution is used, that is, the step size of deconvolution is 2. Insert 0 between its input feature units as a new feature input.

First, the label of the data is encoded by one-hot, and the four fault categories of the rolling bearing fault signal are marked as a four-dimensional vector y, as shown in Table 1.

Table 1 Rolling bearing fault data label one-hot code

Data category	Data encoding
Normal	[1,0,0,0]
Outer ring fault	[0,1,0,0]
Inner ring fault	[0,0,1,0]
Roller fault	[0,0,0,1]

In the generator model shown in Fig. 1, a 100-dimensional random sampling vector z is spliced together with a label encoding vector y as an input layer. The feature is then mapped to 1024 dimensions through a full layer of connections and then BN layer and activated via the ReLU activation function[29] as an output h_0. After h_0 is spliced with the label vector y, the full connection is mapped to 32768 dimensions, and passed through the BN layer[30] and activated by the ReLU activation function as an output h_1. Transform h_1 into a matrix form with dimensions $[16, 16, 128]$. Then h_1 and the label vector are spliced together in the channel dimension to form a size of $[16, 16, 132]$, and $[32, 32, 128]$ is obtained by deconvolution, and activated by the BN layer and the ReLU activation function to obtain h_2. Finally, h_2 to give $[64, 64, 1]$ by deconvolution, and using the tanh activation function to activate as the final output.

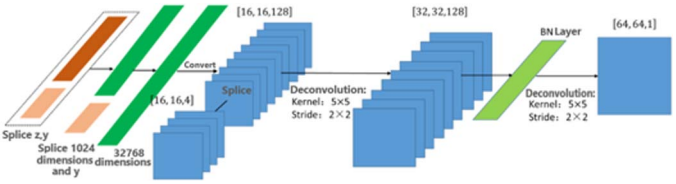

Fig. 1 CWGAN generator model

In the discriminator model shown in Fig. 2, the input is the splicing of the original image and the label vector y in the channel dimension, and the size is $[64, 64, 5]$. With 5×5 convolution kernel, and 2×2 stride, with 0 padding, the output of $[64, 64, 5]$ is obtained and activated using the LeakyReLU activation function. The output and the label vector y are spliced in the channel dimension, the size $[64, 64, 9]$,and the output of $[64, 64, 68]$ is obtained by 5×5 convolution kernel, 2×2 stride, and padding with 0. The output is activated by a Layer BN layer and the LeakyReLU activation function. The upper layer output is transformed into a vector and spliced with the label vector y. After a layer of full connection mapping to 1024 dimensions, pass through a BN layer and activate with the LeakyReLU activation function. The output is spliced with the label vector y, and the full connection is mapped to the 1-dimensional scalar as the final output of the discriminator.

A. *Data preprocessing*

Next, the rolling bearing data is experimentally based on the Morlet-based function wavelet transform. The experimental data is the CWRU rolling bearing fault data set. A continuous wavelet transform method based on Morlet basis function is adopted. Set the scale interval to [1,256] and the step size to 1 for a total of 256 scales. A continuous 512 points are acquired in the original signal as raw data, and wavelet transform is performed. The time domain signals for normal state, inner ring

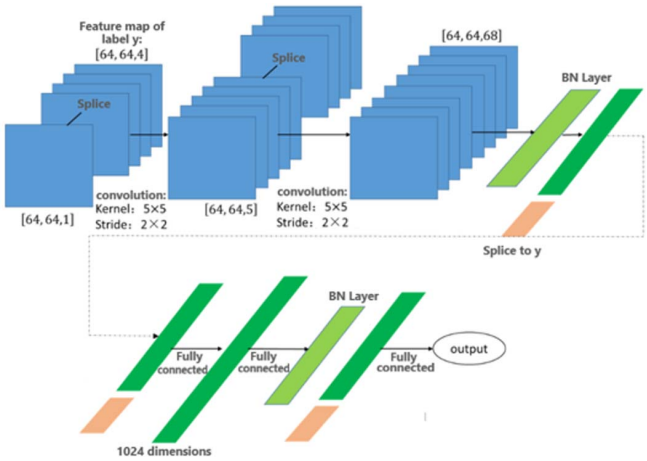

Fig. 2 CWGAN discriminator model

III. EXPERIMENT AND RESULT

fault, outer ring fault and ball fault are analyzed below, and continuous wavelet transform is carried out to draw wavelet time-frequency diagrams of three fault levels under the working condition of load pressure of 0 hp.

(1) Normal state

As can be seen from Fig. 3, in the Morlet wavelet variation time-frequency diagram, there is an energy distribution between the low scale (1 to 50), which has the phenomenon of energy divergence.

(2) Inner ring fault status

Fig. 4 is a time-frequency diagram of the inner ring fault with varying degrees from top to bottom. As can be seen from Fig. 3.2, the energy distribution of the impact signal is most significant at a scale of about 120.

(3) Outer ring fault status

Fig. 5 is a time-frequency diagram of the outer ring faults from top to bottom. As can be seen from the analysis in Fig. 3.3, the energy distribution in the interval [70,200] is more obvious. When the degree of failure is serious, the low-scale interval also has a certain energy distribution.

(4) Ball fault status

Fig. 6 shows the time-frequency diagram of ball failures from top to bottom. As can be seen from the analysis in Fig. 3.4, he wavelet time-frequency signal has a significant periodic impact in the time dimension. Analysis of the wavelet time-frequency diagram in the scale dimension shows that the ball fault is most concentrated at the scale [100, 150], and there is also an energy distribution between [4, 10]. Other scale intervals also have some divergent energy caused by noise.

B. GAN image generation experiment

The wavelet analysis method is used to generate the wavelet time-frequency diagram of the rolling bearing fault. The generated original wavelet time-frequency map size is 256×512. Due to the large size, the wavelet time-frequency map is down sampled by the average pooling method. As shown in Fig. 7, the average pooling uses a kernel of 4×8, with a stride of 4×8, and the image size after downsampling is 64×64.

We normalize the wavelet time-frequency diagram to [-1, 1] through batch normalization.

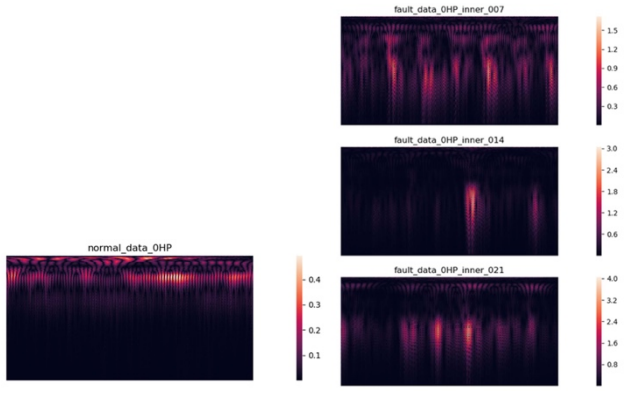

Fig. 3 Wavelet time-frequency diagram of the rolling bearing in normal state

Fig. 4 Bearing inner ring fault wavelet time-frequency diagram

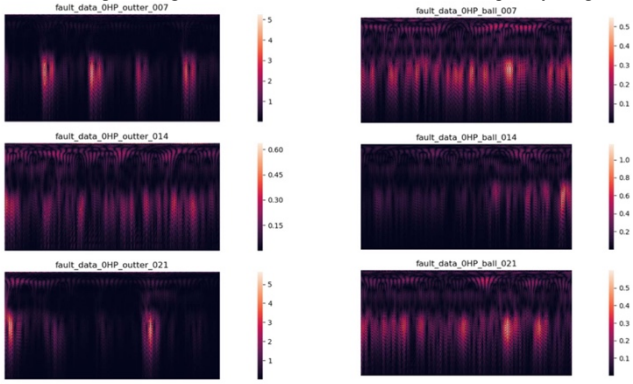

Fig. 5 Bearing outer ring fault wavelet time-frequency diagram

Fig. 6 Bearing ball fault wavelet time-frequency diagram

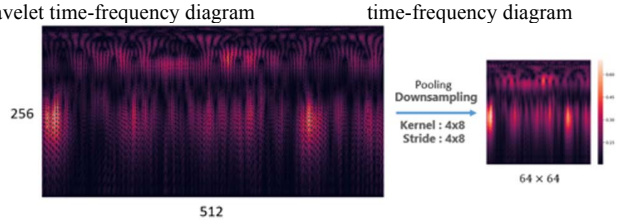

Fig. 7 Rolling bearing fault time-frequency diagram downsampling

A shallow fully connected neural network was used as a generator and discriminator to conduct preliminary experiments to verify the effectiveness of the generated network. The generator input is a 100-dimensional vector z, and z is a Gaussian distribution sample from which the prior distribution p_z is set to a mean of 0 and a variance of 1. In addition to the input layer, the generator of GAN consists of a four-layer fully connected network with 256, 512, 1024, and 4096 neurons. The output layer uses the tanh activation function, and the rest uses the LeakyRelu activation function.

The input neuron of the discriminator is 4096, followed by a three-layer fully connected layer, and the number of neurons in each layer is 1024, 512, 256 and 1. The hidden layers are activated by LeakyReLU, and the last layer uses the sigmoid activation function to map the output to the $[0, 1]$ interval as a probability of determining the input as a real image.

The GAN model is trained by the Adam optimizer[30], and in the training, β_1 ，β_2 are set to 0.9 and 0.999, and ϵ is set to 10^{-8} 。The learning rate is 0.00005 and 5,000 epoch training. The training result shows that the loss function of the generator does not converge, and the error loss gradually increases in the later stage of training. The overall training effect is ineffective.

Use the trained GAN model to generate the wavelet time-frequency diagram. Fig. 8(a) is the image generated by the generator at the beginning of the GAN network training, and Fig. 8(b) is the result in the end. From the results of the generated image, it is known that the training for GAN has an effect of improving the fidelity of the image generated by the generator. However, the quality of the images generated during the post-training period is still low, and the noise is larger than the real wavelet image, as shown in Fig. 9. At the same time, the generated wavelet time-frequency image is more singular, resulting in model collapse.

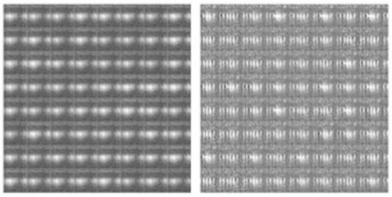

Fig. 8 Spectrum image generated by GAN generator

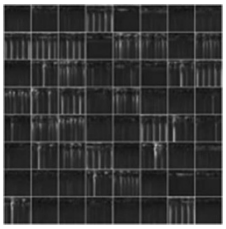

Fig. 9 True wavelet image of the rolling bearing fault

In summary, the above GAN has problems such as training difficulty and unstable training. This is due to the fact that the loss function of the original GAN is not reasonable enough.

C. Analysis of WGAN-GP experiment

For the sake of comparison, the same network structure as above was used for training in the experiment. The prior distribution p_z is set to a Gaussian distribution with a mean of 0 and a variance of 1. The rolling bearing fault data was put into the network for training. The learning rate α was set to 0.00005, optimized by the Adam optimizer. The batch size of the training input is 64, and 100 epochs was trained. The training results show that the generator and discriminator training have large oscillations in the early stage of training, but as the training progresses, the oscillation gradually stabilizes and the training loss function gradually converges. The overall generator and discriminator training is relatively stable.

The wavelet time-frequency map is generated by using the trained WGAN-GP model. Fig. 10(a) is the image generated by the generator after generated by the first epoch after generative adversarial network training and Fig. 10(b) is generated after 10th epoch. Fig. 10(c) is generated after the 30th epoch. From the results of the generated images, it can be seen that the WGAN-GP generated images are greatly improved compared to GAN and WGAN. From the perspective of the generated image quality, it has been close to the real image at about 3000 epochs. From the diversity of images, it can be seen that wavelet time-frequency maps of various categories can be generated in 64 samples, and the overall is effective.

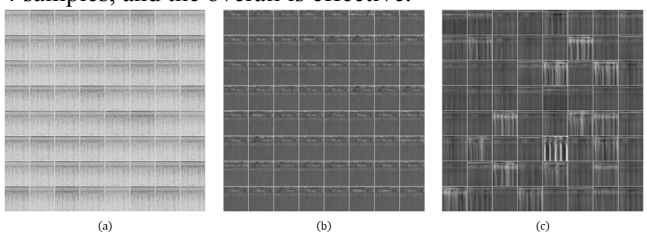

Fig. 10 Spectrum image generated by WGAN-GP generator

D. Analysis of CWGAN experiment

In order to verify the effect of CWGAN, the distribution of random noise samples is set to a Gaussian distribution with a mean of 0 and a variance of 1. The wavelet time-frequency map is generated by the rolling bearing fault data and put into the network for training. The learning rate is set to 0.00005, optimized with the Adam optimizer, training batch size of 64 for one input, and training 100 epoch. Model parameter initialization is randomly generated in a truncated Gaussian distribution with a mean of 0 and a variance of 0.02. The truncated Gaussian distribution is: if the generated value is greater than the value of the standard deviation of 2 standard deviations, the reselection is discarded.

Using the well-trained GAN network model to generate the wavelet time-frequency map, and each category samples 8 pictures. The GAN network generated image is normalized in the [-1, 1] interval image, so the normalized image needs to be restored, as shown in Fig. 11.

From the perspective of generating images, the effect of the control category is achieved by adding the condition vector y to the input. By adopting the idea of DCGAN and WGAN-GP, the quality of the image generated by the model is improved, and the generated image is close to the real wavelet time-frequency diagram.

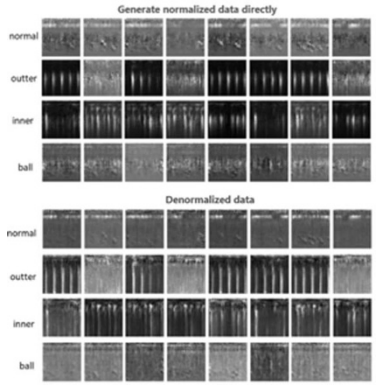

Fig. 11 Wavelet time-frequency diagram of CWGAN generator

E. CWGAN training effect verification

The following is to verify the complementary effect of the GAN generated image. We use 100, 200, and 300 original wavelet time-frequency maps as the training set. In order to control the variables, the training set is used to train CWGAN, and 2048 images are generated by using the trained CWGAN. CWGAN generated image is supplemented as data to train the classic deep learning network VGG-Net[31]. The input layer passes through three layers of convolutional layers, the number of convolution kernel channels is 32, 64, 128, and the convolution kernel size is 3×3, the stride is 1. Then pass through a max pooling layer, the kernel is 2×2 and the stride is 2, and the fully connected layer is mapped to the 512-dimensional feature, and finally the 4-dimensional vector is output. A total of 30 epochs were trained.

Fig. 12 shows the accuracy of the test set when the training set is 100 original wavelet time-frequency diagrams without adding CWGAN generated data (a) and adding CWGAN generated data (b) as the training process changes. It can be seen that when the CWGAN data is not added, because of the amount of data in the training set is too small, VGG is easy to fall into

the over-fitting situation. And the final accuracy is about 88%. After adding CWGAN generated data as supplementation, the over-fitting phenomenon is alleviated, and finally converges at about 95%.

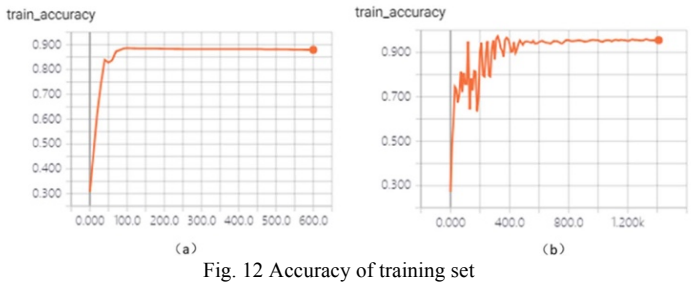

Fig. 12 Accuracy of training set

Table 2 shows the experimental results. It can be seen that the network identification accuracy rate is 88%, 97%, and 98% when generated data is not added. After the data generated by CWGAN is added for training, VGG-Net increased the accuracy by 6.6%, 2.3%, and 1.2% respectively. Therefore, the experiment proves that the data supplementation through CWGAN network can effectively reduce the over-fitting effect of deep learning model caused by the lack of data. And the smaller the training data volume, the more obvious the effect of CWGAN generated data on model training improvement.

Table 2 CWGAN data supplementation experiment results

The amount of training set data	The amount of test set data	The amount of CWGAN generated data	Accuracy: add CWGAN generated data as supplement	Accuracy: no data supplement	Increase amount
100	7291	2048	0.9525	0.8858	**0.0667**
200	7191	2048	0.9727	0.9496	**0.0231**
300	7091	2048	0.9859	0.9737	**0.0122**

IV. Conclusion

In actual industrial scenarios, fault data often has the characteristics of insufficient data. Fault diagnosis based directly on the deep learning model often results in model overfitting due to less fault data. This paper proposes an innovative deep learning model: CWGAN. CWGAN can learn the characteristics of fault data and generate data close to the real fault data, which can supplement the fault data with insufficient data volume, and improve the accuracy of fault diagnosis in actual industrial scenarios.

CWGAN combines the design idea of CGAN and DCGAN, and uses WGAN-GP training method. So CWGAN can control the type of generated data by inputting the label, and at the same time ensure the fidelity of the generated data. Comparing the training effect of the fault data training set with the generated data on VGG-Net, it is found that the recognition accuracy is improved by 3% after adding the generated data by CWGAN, which effectively proves the complementary effect of CWGAN on the fault data with insufficient data, and avoids the problem of over-fitting in the training of deep learning.

References

[1] Lv F, Wen C, et al. Fault diagnosis based on deep learning [C]. American Control Conference. American Automatic Control Council (AACC), 2016.

[2] Junbo T, Weining L, Juneng A, et al. Fault diagnosis method study in roller bearing based on wavelet transform and stacked auto-encoder [C]. Control & Decision Conference. IEEE, 2015.

[3] Lecun Y, Bengio Y. Deep learning [J]. Nature, 2015, 521(7553):436.

[4] Deng L, Yu D. Deep learning: methods and applications [J]. Foundations & Trends in Signal Processing, 2014, 7(3):197-387.

[5] Jiang H. Face Detection with the Faster R-CNN [J]. International Conference on Automatic Face & Gesture Recognition, 2017:650-657.

[6] Redmon J, Divvala S, Girshick R, et al. You Only Look Once: Unified, Real-Time Object Detection [J]. 2015.

[7] Shaobo L, Guokai L, Xianghong T, et al. An Ensemble Deep Convolutional Neural Network Model with Improved D-S Evidence Fusion for Bearing Fault Diagnosis [J]. Sensors, 2017, 17(8).

[8] Duong B P, Kim J M. Non-Mutually Exclusive Deep Neural Network Classifier for Combined Modes of Bearing Fault Diagnosis [J]. Sensors, 2018, 18(4):1129.

[9] Qi Y, Shen C, Jie L, et al. An Automatic Feature Learning and Fault Diagnosis Method Based on Stacked Sparse Autoencoder [C]. International Workshop of Advanced Manufacturing & Automation, 2017.

[10] Sun W, Shao S, Rui Z, et al. A sparse auto-encoder-based deep neural network approach for induction motor faults classification [J]. Measurement, 2016, 89(ISFA):171-178.

[11] Yuan M, Wu Y, Lin L. Fault diagnosis and remaining useful life estimation of aero engine using LSTM neural network [C]. IEEE International Conference on Aircraft Utility Systems. IEEE, 2016.

[12] Vincent P.et al. Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion [J]. Journal of Machine Learning Research, 2010, 11(12):3371-3408.

[13] Karpathy A.et al. Large-Scale Video Classification with Convolutional Neural Networks [C]. 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE Computer Society, 2014.

[14] Hinton G E. Deep belief networks [J]. Scholarpedia, 2009, 4(6):5947.

[15] Hansen L K. Neural Network Ensembles [J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2002, 12(10):993-1001.

[16] Ledig C, Theis L, et al. Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network [C]. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2017.

[17] Inoue H. Data Augmentation by Pairing Samples for Images Classification [J]. 2018.

[18] Fawzi A, Samulowitz H, Turaga D, et al. Adaptive data augmentation for image classification [C]. 2016 IEEE International Conference on Image Processing (ICIP). IEEE, 2016.

[19] Almahairi A, Rajeswar S, Sordoni A, et al. Augmented CycleGAN: Learning Many-to-Many Mappings from Unpaired Data [J]. 2018.

[20] Zhu X, Liu Y. Data Augmentation in Classification using GAN [J]. 2017.

[21] Chen Wei. Application of deep learning in fault diagnosis of rolling bearings [D]. Southwest Jiaotong University, 2018.

[22] Dong S, Zhang Z, Wen G, et al. Design and application of unsupervised convolutional neural networks integrated with deep belief networks for mechanical fault diagnosis [C]. 2017 Prognostics and System Health Management Conference (PHM-Harbin). IEEE, 2017.

[23] Schniter P. Short-time Fourier Transform [J]. in Advanced Topics in Signal Processing, J.S. Lim and A.V Oppenheim, 2009, 32(2):289-337.

[24] Daubechies I. The wavelet transform, time-frequency localization and signal analysis [J]. Journal of Renewable & Sustainable Energy, 1990, 36(5):961-1005.

[25] Arjovsky M, Chintala S, Bottou L. Wasserstein GAN [J]. 2017.

[26] Gulrajani I, Ahmed F, Arjovsky M, et al. Improved Training of Wasserstein GANs [J]. 2017.

[27] Mirza M, Osindero S. Conditional Generative Adversarial Nets [J]. Computer Science, 2014:2672-2680.

[28] Radford A , Metz L , Chintala S. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks [J]. Computer Science, 2015.

[29] Fergus R, Fergus R, Fergus R, et al. Deep generative image models using a Laplacian pyramid of adversarial networks [C]. International Conference on Neural Information Processing Systems, 2015.

[30] Kingma D P, Ba J. Adam: A Method for Stochastic Optimization [J]. Computer Science, 2014.

[31] Simonyan K, Zisserman A. Very Deep Convolutional Networks for Large-Scale Image Recognition [J]. Computer Science, 2014.