Mechanical Systems and Signal Processing 185 (2023) 109772

Contents lists available at ScienceDirect

Mechanical Systems and Signal Processing

journal homepage: www.elsevier.com/locate/ymssp

Deep convolutional generative adversarial network with semi-supervised learning enabled physics elucidation for extended gear fault diagnosis under data limitations

Kai Zhou [a], Edward Diehl [b], Jiong Tang [c,*]

[a] Department of Mechanical Engineering-Engineering Mechanics, Michigan Technological University, USA
[b] Department of Mechanical, Aerospace, and Acoustical Engineering, University of Hartford, USA
[c] Department of Mechanical Engineering, University of Connecticut, Storrs, CT 06269, USA

ARTICLE INFO

Communicated by John E. Mottershead

Keywords:
Gear fault diagnosis
Semi-supervised learning
Deep convolutional generative adversarial network (DCGAN)
Scarce labeled data

ABSTRACT

Fault detection and diagnosis of gear systems using vibration measurements play an important role in ensuring their functional reliability and safety. Computational intelligence, leveraging upon classification through various surrogate models, has recently demonstrated certain level of success. Major challenge however remains. The establishment of surrogate models generally requires large size of training data with specific labels corresponding to explicitly known gear fault conditions, which may not be available in practical applications. Both the size of available data and the respective labels may be quite limited due to the high cost, which hinders the diagnosis of unseen/unexpected faults with desired reliability. In this research we synthesize a deep convolutional generative adversarial network (DCGAN) to tackle this challenge. This new approach follows the semi-supervised learning concept, the performance of which is significantly enhanced by introducing additionally the inexpensive unlabeled data. The balanced adversarial effect between the discriminator and generator in DCGAN is realized by appropriately designing their architectures, which as a result can enable the high accuracy of diagnosis with scarce labeled data. More importantly, by taking full advantage of the rich fault signatures in the unlabeled data that point to the diverse unseen faults, the intrinsic correlation of underlying physics between the unseen and known faults can be implicitly elucidated via unique semi-supervised learning strategy featured in DCGAN. Therefore, the extended capability in diagnosing the unseen faults that are beyond the known faults in training dataset can be realized, which bears practical significance. Systematic case studies using experimental data acquired from a lab-scale gear system are carried out to validate the new diagnosis framework.

1. Introduction

As one type of important components in rotating machinery, gearboxes are widely used in vehicle transmission, various turbines, and manufacturing equipment [1]. The gearbox usually is subjected to complex and harsh operation conditions, such as constant and heavy dynamic load, which may cause the occurrence of fault conditions. Condition monitoring and fault diagnosis of the gearbox therefore are essential for ensuring the system reliability and safety. The gear fault diagnosis generally is implemented based upon the

* Corresponding author.
E-mail address: jiong.tang@uconn.edu (J. Tang).

https://doi.org/10.1016/j.ymssp.2022.109772
Received 14 January 2022; Received in revised form 27 May 2022; Accepted 4 September 2022

different types of signals collected, including vibration [2], acoustic emission [3] and eddy current measurements [4]. Among them, the vibration signals are intrinsically rich, and can be easily acquired through the usage of low-cost sensors that are embedded. Vibration analysis allows one to leverage advanced signal processing and feature extraction techniques. For instance, the raw time-domain vibration signals can be converted into the time–frequency representation using wavelet transforms. Various wavelet transform methods have been reported in literature [5,6], demonstrating their feasibility. However, the fault diagnosis performance of these methods varies with respect to the specific feature selection which to a large extent is empirical.

Computational intelligence has recently become an emerging technology for gear fault diagnosis owing to the rapid advancement of computing power and resources. The underlying concept of computational intelligence is to explore the intrinsic relation between the fault conditions and vibration signals using so called surrogate models. Fundamentally, condition monitoring and fault diagnosis of gear systems are performed via classification analysis, in which the surrogate models function as classifiers. The surrogate model-based gear fault diagnosis possesses high flexibility and can be applied in conjunction with the aforementioned signal processing techniques [7–9]. On the other hand, deep learning neural networks, as one particular type of surrogate models, have the potential to avoid manual feature extraction which is empirical and subjective. When sufficiently large amount of data is involved for the training, deep neural networks can characterize the data feature correlation accurately due to their powerful inference capacity. Unlike the general surrogate models, the desirable performance of the deep learning neural networks can be achieved without the need of implementing data pre-processing, since different types of layers integrated allow one to automatically extract and learn the features at different hierarchical levels. Sun et al proposed the particle swarm optimization to perform the automatic feature extraction and then integrated the particle swarm optimization into the network to further identify gearbox composite faults [10]. You et al applied the deep convolutional neural network with improved rectified linear activation function into machinery fault diagnosis directly utilizing the raw vibration signals [11]. Chen et al developed an adaptive neural network with respect to the rotating speed to achieve the gear fault diagnosis under non-stationary conditions, the performance of which has been validated through the experimental investigation [12]. To address the "black-box" issue of conventional deep learning neural networks, the physically interpretable/explainable deep learning neural networks recently have received growing attention in machinery condition diagnosis [13–17]. The basic idea of these neural networks is to integrate physics-based signal processing technique into the network layers, from which the physics-related fault features will be visualized and their effect on the decision-making process during the training will be explained from a physical point of view.

While deep learning methods are promising for autonomous fault diagnosis of gears, their performance hinges upon the size and quality of the training data. In actual operation of the gearbox system, fault occurrence is relatively infrequent, and faults in gear systems are continuous in nature with very large (infinite in theory) number of profiles. Therefore, the acquisition of large amount of high-quality data, i.e., labeled training samples corresponding to various fault types, is extremely costly and may not be feasible for deep learning. To address the challenge of training data size and quality, recently the transfer learning approach in which the deep learning methods are used as backbone has been employed in machinery fault diagnosis [18–21]. The transfer learning approach can alleviate the overfitting under small dataset and in some cases lead to good accuracy. In addition to transfer learning, the physics-informed neural networks (PINNs) have recently emerged as a new type of approaches for conducting machinery fault diagnosis [22–25]. Compared to purely data-driven neural networks, the PINNs that incorporate the physical law/knowledge into the networks can provide additional constraints to the network outputs, alleviating the overfitting issues, reducing the need of big training datasets, and thus improving the model robustness for reliable fault prediction. Because of the built-in physical law, the PINNs also have the flexibility to become fully or partially interpretable. Moreover, by leveraging the domain knowledge, the root cause of incorrect diagnosis results can be elucidated from a physical perspective.

As can be seen in above literature, the progresses on the fault diagnosis using limited high-quality data, i.e., labeled data indeed have been achieved to certain extent. However, the transfer learning-based deep learning may possess unsatisfactory performance when the improper domain knowledge adaption for different problems is adopted. The PINN on the other hand needs to be developed/designed according to the particular problem of interest. In other words, the physical law and the way to embed the physical law vary case by case. More importantly, the fault diagnosis using these types of methods has been implemented as supervised tasks, in which the other resources except the labeled data are not fully utilized. It is worth noting that acquiring unlabeled data directly from in-service gearbox systems are generally inexpensive as compared to the acquisition of labeled data. Some recent efforts in computational intelligence have suggested that a semi-supervised learning scheme that leverages upon unlabeled data which carry implicitly important features of the gear systems could enhance the inference performance. Among the semi-supervised learning methods, generative models are the mainstream of the methods built upon the Bayes' rule, aiming at estimating the distribution of data points that belong to each fault class [26]. With the advancement of deep learning, deep generative models (DGMs), as a new class of generative models is formed through the combination of generative models and deep neural networks [27]. Generative adversarial network (GAN) certainly is a viable variant of DGMs, which has been practiced in diverse scientific domains [28–30].

In general, the unseen or unexpected faults in actual fault diagnosis practice are far beyond the limited fault labels available in training dataset [31]. Therefore, achieving the extended fault diagnosis based upon the limited known fault labels is of significant importance. Instead of adding the *explicit* physical law into the network to construct the aforementioned physics-informed neural network (PINN), in this research we intend to establish a different path for enhanced fault diagnosis by exploiting the *implicit* physical correlation between the faults. Specifically, we leverage the unique semi-supervised learning of GAN that can take both the limited labeled vibration signals and the rich unlabeled signals into account during the training. Usually, the large amount of unlabeled data can be inexpensively acquired from the in-service machinery system. The large-sized unlabeled data likely contain very rich fault signatures that point to many unknown/unseen faults, which can be further intrinsically linked to the signatures of known faults in the labeled data via semi-supervised learning/training. This has the potential to not only overcome the data limitation challenge and

alleviate the data acquisition cost, but also may enable the implicit elucidation of physical correlation between the known and unknown/unseen fault labels, which is promising for the extended fault diagnosis mission. In this research, to streamline the entire fault diagnosis process, the deep learning features are incorporated into the traditional GAN to establish a deep convolutional generative adversarial network (DCGAN). With the built-in powerful feature extraction function in DCGAN, the fault diagnosis can be directly performed upon the raw vibration signals in the time domain.

The remainder of this paper is organized as follows. In Section 2, the gear fault diagnosis framework built upon DCGAN is outlined, in which the intrinsic interaction between the discriminator and generator in DCGAN is explained during model establishment followed by the introduction of extended fault diagnosis using the well-established DCGAN. Section 3 provides the methodology formulation and implementation on the fault diagnosis practice of the gearbox system with laboratory setting. Concluding remarks are summarized in Section 4.

2. DCGAN-Based gear fault diagnosis framework

In this section, we outline the fundamental idea of DCGAN-based framework for gear fault diagnosis. Since the GAN is the mainstay of the DCGAN, we will use GAN for the presentation throughout this section unless otherwise specified. The original generation of GAN firstly proposed by Goodfellow et al [32] was designed to fit the unsupervised learning purpose. This model consists of a generator and a discriminator. The goal of the generator is to produce the plausible synthetic data by incorporating the feedback from the discriminator. The fake data produced via generator attempt to deceive the discriminator. The discriminator essentially is a binary classifier, trying to distinguish the real data from the fake data created by the generator. Two networks are trained synchronously with close interaction, resulting in the overall performance enhancement of GAN. One important application of the unsupervised learning GAN is the image reconstruction through the generator [33–35]. Such idea also has been executed on the machinery fault data augmentation and enrichment [36–38]. Unfortunately, the unsupervised learning GAN still is limited to the small range of applications because it lacks the notion of classes. To accommodate a wide variety of applications by taking full advantage of the unlabeled data, the semi-supervised concept has been recently adopted to extend the GAN, enabling the discriminator to perform as both binary and multi-class classifiers/discriminators proposed by Salimans et al [39].

Assume the generator is denoted as G, and the discriminator is denoted as D. G and D are the competitors in a minmax game. The cost of this game can be mathematically described as [40],

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

where E is the empirical estimate of expected value of the probability. D attempts to distinguish between the real data that is represented by true distribution $p_x(x)$ and the fake data that is generated by G. G transforms a random noise vector from the latent space into $G(z)$, which is a sample from the distribution $p_z(z)$, and ideally the distribution $p_z(z)$ will converge to the true distribution $p_x(x)$. The min and max functions in Equation (1) refer to the minimization of $\log(1-D(G(z))$ (or maximizing the generator loss i.e., $-\log(1-D(G(z)))$ and maximization of $\log(D(x)) + \log(1-D(G(z))$(or minimizing the discriminator loss, i.e., $-\log(D(x)) - \log(1-D(G(z)))$. Specifically, the discriminator seeks to maximize the differentiation ability of real/true and fake samples if the discriminator is a binary classifier, whereas the generator seeks to minimize the odds of fake sample detection by the discriminator, promoting the generator to produce samples that have a low probability of being fake. One may notice that there is adversarial effect existing on the term $\log(1-D(G(z))$ due to the minmax game theory, which is the unique feature of the GAN.

In semi-supervised learning, the fault labels are not available for all training samples. Therefore, it is essential to leverage unlabeled data for estimating an appropriate prior to be used by the classifier. In this research, we adopt the extended GAN, in which the integrated discriminator model, consisting of both a multi-class classifier and a binary classifier, is employed to learn the prior distribution. The discriminator can justify whether a sample belongs to the true distribution (real data) or not, and meanwhile determine what fault label that a sample belongs to. Correspondingly, two types of losses representing the binary and multi-class classification accuracy, respectively, are defined to train the discriminator. The training performance of a binary classifier is assessed by so called binary cross-entropy. Its form upon one single training sample is mathematically formulated as

$$L_{D_b} = -y_T\log(\widehat{y}_T) - (1-y_T)\log(1-\widehat{y}_T) \tag{2}$$

where y_T denotes the true sample when taking a value 1, and $\widehat{y}_T$ is the probability of a sample to be true sample, which can be realized by applying the softmax activation function to the output layer. $y_T = 0$ on the other hand, denotes the fake sample, and $1-\widehat{y}_T$ thus represents the probability of fake sample. Since the unlabeled data essentially are real, the respective binary cross-entropy loss can be simplified as

$$L^T_{D_b} = -\log(\widehat{y}_T) \tag{3}$$

It is noted that the binary classifier in the GAN also needs to differentiate the fake samples produced from generator with labels $y_T = 0$. The respective binary cross-entropy can be described as

$$L^F_{D_b} = -(1-y_T)\log(1-\widehat{y}^G_T) = -\log(1-\widehat{y}^G_T) \tag{4}$$

where $\widehat{y}^G_T$ indicates the probability of a fake sample produced via generator to be true sample identified by binary classifier. Therefore, the general binary cross-entropy to evaluate the performance of binary classifier is the sum of $L^T_{D_b}$ and $L^F_{D_b}$,

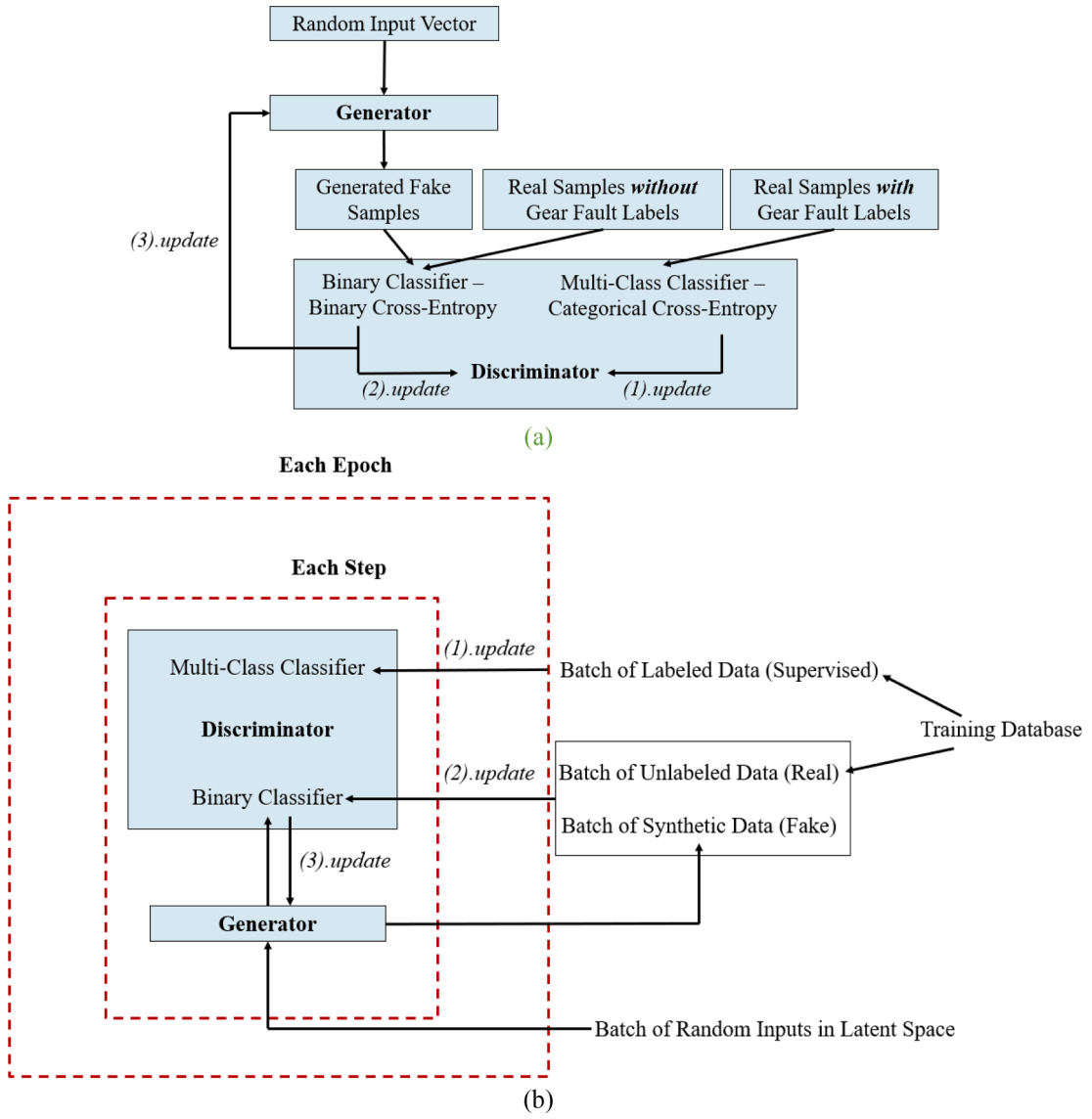

Fig. 1. GAN model training (a) overall training framework; (b) batch training framework.

$$L_{D_b} = -\log(\widehat{y}_T^{D_b}) - \log(1 - \widehat{y}_T^{G}) \tag{5}$$

For the supervised-learning purpose, the multi-class classification analysis for fault condition identification also is conducted according to the categorical cross-entropy loss that is defined as [41].

$$L_{D_c} = -\sum_{i=1}^{n} y_i \log(\widehat{y}_i) \tag{6}$$

where y_i is the true class label and $\widehat{y}_i$ is the probability of a sample to be the class y_i. n generally is the number of classes. In this research, n is the number of known fault labels in a multi-class classifier. Since only one label is considered true in multi-class classification analysis, the simplified loss can be obtained as,

$$L_{D_c} = -\log(\widehat{y}_i^{D_c}) \tag{7}$$

where superscript D_c denotes the multi-class classifier for classifying the real samples to be different fault labels. Hence, the total loss of the discriminator in GAN can be described as

$$L_D = L_{D_b} + L_{D_c} = -\log(\widehat{y}_T^{D_b}) - \log(1 - \widehat{y}_T^{G}) - \log(\widehat{y}_i^{D_c}) \tag{8}$$

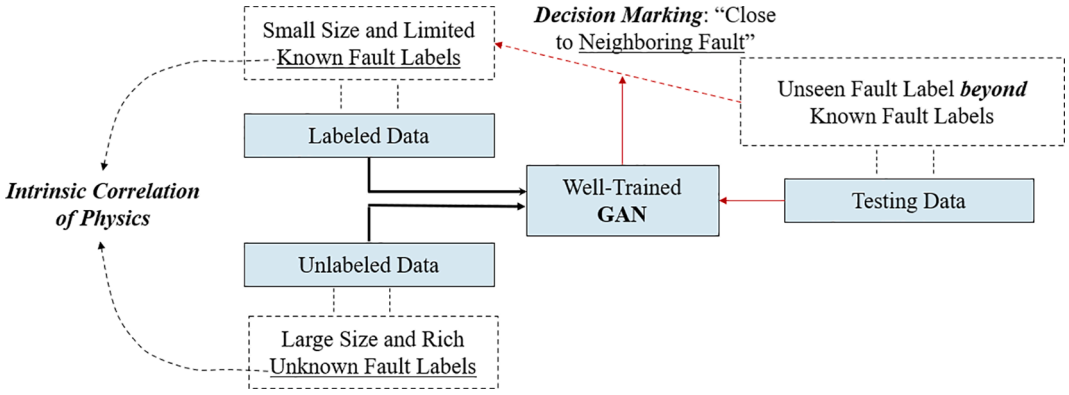

Fig. 2. Extended Fault Diagnosis using GAN.

The training of the discriminator aims at minimizing the discriminator loss (Equation (8)). It is worth noting that, the loss of generator also is evaluated by the binary classifier. The difference of the generator loss as compared with the loss of binary classifier shown in Equation (5) is that it only is related to the fake sample. The generator loss needs to be maximized, indicating that the generator is smart enough to deceive the binary classifier. The loss of generator thus can be defined as

$$L_G = -\log(1-\widehat{y}_T^G) \tag{9}$$

To facilitate the model training, we convert the generator loss (Equation (9)) to a new form,

$$L_G = -\log(\widehat{y}_T^G) \tag{10}$$

Such conversion renders the generator loss to be minimized as well via training.

The cost functions represent the expected values/mean values of defined losses given in Equations (8) and (10), which can be mathematically expressed as

$$C_D = -\frac{1}{M}\sum_{i=1}^{M}(\log(\widehat{y}_T^{D_b}) + \log(1-\widehat{y}_T^G) + \log(\widehat{y}_i^{D_c})) \tag{11a}$$

$$C_G = -\frac{1}{M}\sum_{i=1}^{M}(\log(\widehat{y}_T^G)) \tag{11b}$$

where C_D denotes the cost function of the discriminator, which is sum of the cost functions of binary and multi-class classifiers. C_G denotes the cost function of the generator. As will be shown, the batch training is adopted. M hence is the batch size. Equation (11) can be further re-written as

$$C_D = -E(\log(\widehat{y}_T^{D_b})) - E(\log(1-\widehat{y}_T^G)) - E(\log(\widehat{y}_i^{D_c})) \tag{12a}$$

$$C_G = -E(\log(\widehat{y}_T^G)) \tag{12b}$$

Minimizing the cost functions in Equation (12) fundamentally points to the minmax game theory indicated in Equation (1).

The overall training framework of GAN is illustrated in Fig. 1a, indicating the collaborative learning process of both the discriminator and the generator. It is worth noting that, the entire training is generally performed in a nested structure, in which each batch of each epoch will be considered as a unit step that contributes to the network parameter updating (Fig. 1b). Here, the sizes of epochs and batches are two hyperparameters to be specified beforehand. The total number of unit steps is equal to the total training data size divided by the batch size. The training database is prespecified, which can provide the batches of data with different types to update the corresponding models. The only batch data that are not selected from the training database are the inputs of the generator, which instead are built upon the random latent points given the batch size.

Consistent with the overall training framework (Fig. 1a), the batch training strictly follows 3 consecutive steps aiming at minimizing the cost functions defined in Equation (12) as shown in Fig. 1b. In particular, the first step is to train the multi-class classifier by involving the batch of labeled data, aiming at minimizing the cost function associated with categorical cross-entropy loss (i.e., $-E(\log(\widehat{y}_i^{D_c}))$ in Equation (12a)). In the second step, the fake data produced via generator in terms of random noise vector and the real unlabeled data are combined to train the binary classifier, aiming at minimizing the cost function associated with binary cross-entropy loss (i.e., $-E(\log(\widehat{y}_T^{D_b}))-E(\log(1-\widehat{y}_T^G))$ in Equation (12a)). It is worth noting that the binary and multi-class classifiers share the common parameters to be updated because they are embedded into one discriminator. The discriminator hence is trained in both Steps 1 and 2. In the last step, i.e., Step 3, keeping the discriminator obtained from Step 2 fixed, the generator is updated, which converts the random input/noise vectors into the plausible synthetic data. Batch of synthetic data then is substituted into the binary classifier to

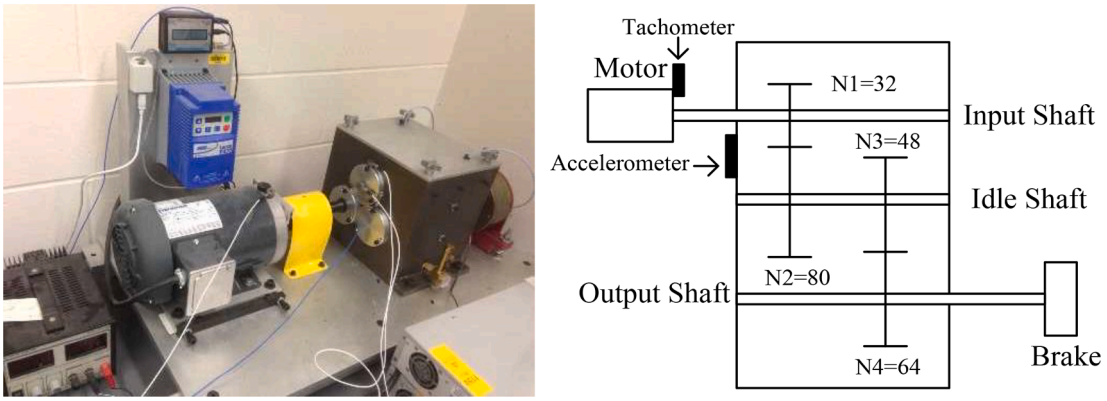

Fig. 3. Gearbox setup for data acquisition.

Fig. 4. 9 fault types on pinions, including 5 chipping tip types with 5 different severities.

evaluate the cost function associated with the generator loss (Equation (12b)), which directs the generator updating. For implementation convenience, the labels of synthetic data are marked as real data in this step to train the generator, while the labels are marked as fake data in the step 2 to train the binary classifier. The label inverting here essentially is due to the mathematical form change of generator loss, i.e., from Equation (9) to (10). The backpropagation optimization strategy is employed for GAN training [35]. Stochastic gradient descent (SDC) [36,37] is a commonly used learning algorithm. One important hyperparameter of this optimization algorithm is the learning rate, which determines the increment size at each iteration while moving toward a minimum of a cost function.

As mentioned, owing to the unique semi-supervised learning scheme of GAN, the applications of GAN can be expanded beyond the regular classification analysis-based gear fault diagnosis. Since the large amount of unlabeled data may implicitly point to many unknown/unseen fault labels beyond the known fault labels contained in the labeled data, the semi-supervised learning through combining these two types of data allows one to characterize the intrinsic correlation of physics between the known and unseen gear faults, thereby fulfilling the extended fault diagnosis as illustrated in Fig. 2. Such practical application in gear fault diagnosis will be illustrated via case studies in the subsequent section.

3. Formulation of fault diagnosis with experimental measurement

In this section, we present, based on the DCGAN framework outlined in the preceding section, actual fault diagnosis implementation and validation using experimental vibration measurement. The details of experimental data acquisition, diagnosis problem setup, the particular DCGAN model development, analysis procedures, and result discussion are presented consecutively.

3.1. Experimental data acquisition and fault labeling

The gear vibration responses usually carry rich signatures, which reflect the health status of a gearbox system. In this study, the experimental investigation is carried out based upon the direct vibration measurement from a lab-scale gearbox system with replaceable gears shown in Fig. 3. A motor is used to control the gear speed following the trapezoidal profile. The torque is applied by a magnetic brake controlled by its input voltage. On the first stage input shaft, a 32-tooth pinion and an 80-tooth gear are installed, while a 48-tooth pinion and a 64-tooth gear are installed on the second stage. A tachometer and an accelerometer are utilized to measure the input shaft speed and gear vibration signals, respectively. The vibration signals are acquired using dSPACE system with 20 KHz

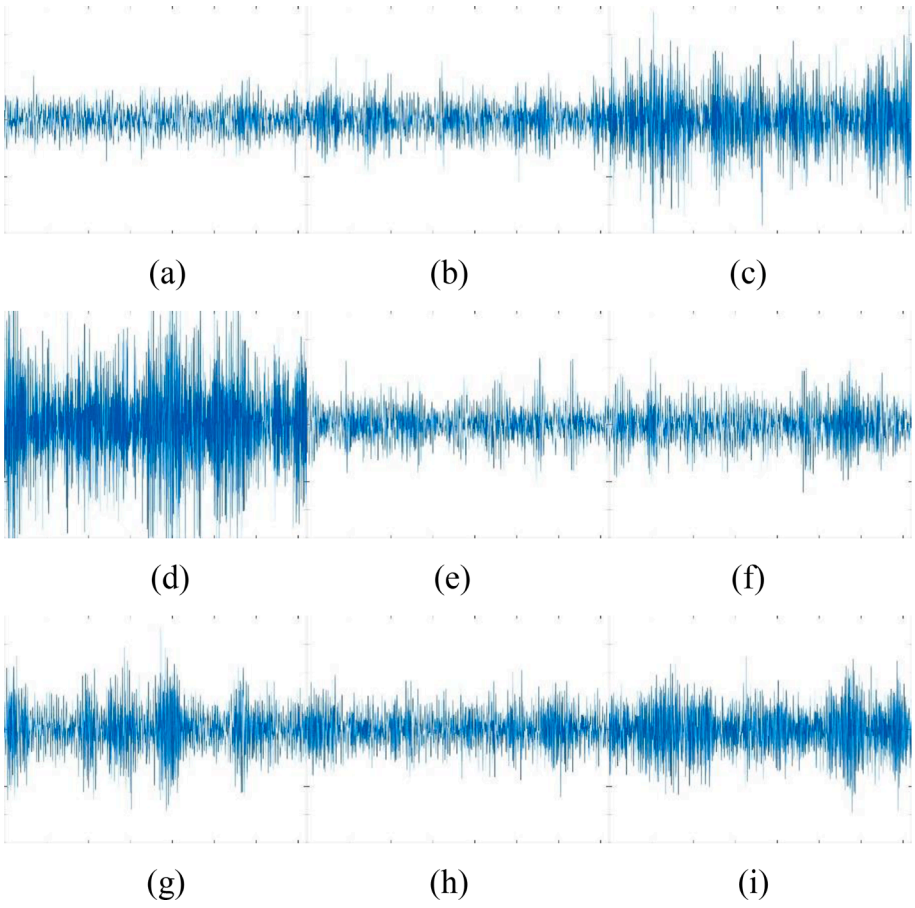

Fig. 5. Gear vibration signal samples (a) healthy; (b) missing tooth; (c) crack; (d) spalling; (e) chipping tip 1 (least severe); (f) chipping tip 2; (g) chipping tip 3; (h) chipping tip 4; (i) chipping tip 5 (most severe).

sampling frequency. In this research, the time synchronous averaging (TSA) approach is employed to minimize the measurement uncertainty and convert the signals from time-even to the angle-even representation, which yields the response with reduced non-coherent components. The validity of the data was analyzed with manual feature extraction [42].

9 different gear conditions are introduced into the pinion on the input shaft, including the *healthy condition*, *missing tooth*, *root crack*, *spalling*, and *chipping tip* with 5 different severities/levels, as shown in Fig. 4. Altogether, we have 9 classes/labels for the data collected. For each gear condition, 104 signals are collected. Hence, we have a total of 936 (104×9) samples corresponding to 9 gear conditions. For each sample of vibration signals, 3,600 angle-even data points are recorded in the course of 4 gear revolutions. In other words, each sample includes 3,600 features. The raw vibration signals of associated gear conditions can be seen in Fig. 5. Except for the gear crack and spalling that have more significant amplitudes in the raw vibration signals, other gear faults contain similar features in the raw vibration signals. Therefore, current features in the vibration signals cannot be directly used to pinpoint the gear faults. To facilitate the accurate gear fault diagnosis, the powerful feature extraction is required. It is noted that the feature extraction function has been already embedded into the DCGAN network. Through extracting the useful features from raw vibration signals gradually along the network depth, the inherent correlation between the raw vibration signals and respective faults can be adequately characterized, thereby facilitating the fault diagnosis practice. It is worth mentioning that the images with entire feature information in the signals are directly used and fed into the DCGAN training. All the data used in this study is made public at [43].

3.2. DCGAN architecture configuration

GAN generally includes a discriminator and a generator. In this research, we incorporate the deep learning features into the GAN to establish a DCGAN, which can facilitate the powerful feature extraction directly based on the vibration signals. As mentioned, we extend the original unsupervised DCGAN learning into the semi-supervised DCGAN learning to achieve the research objective. This requires the discriminator model to be an integration of both the binary and multi-class classifiers. Binary classifier is to differentiate the real and synthetic fake images, whereas the multi-class classifier is to identify the different fault labels, which aligns with the ultimate diagnosis goal. The models embedded in DCGAN are closely and continuously interacted with each other during the training. Therefore, the core of this research is to specifically design the appropriate architectures of the discriminator and generator, upon

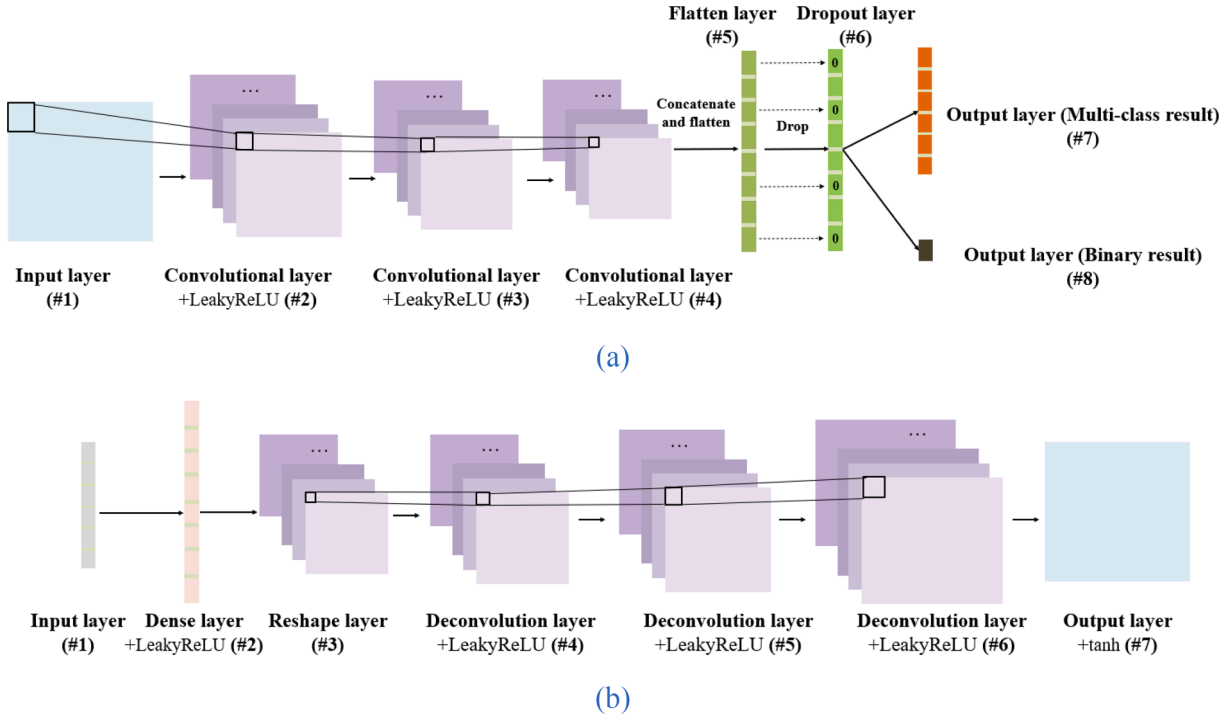

Fig. 6. DCGAN architecture (a) discriminator; (b) generator. (Note: the small frame marked denotes the convolution or deconvolution operation of the filter along the network depth).

Table 1
Discriminator configuration (multi-class and binary classifiers integrated).

Layer ID	Layer Type	Output Shape	Parameters
#1	Input	$64 \times 64 \times 1$	0
#2	Convolutional (LeakyReLU)	$32 \times 32 \times 32$	320
#3	Convolutional (LeakyReLU)	$16 \times 16 \times 64$	18,496
#4	Convolutional (LeakyReLU)	$8 \times 8 \times 128$	73,856
#5	Flatten	4,096	0
#6	Dropout	4,096	0
#7	Output (multi-class)	9	57,351
#8	Output (binary)	1	0

Note: 3×3 convolutional filters with 128 channels, zero-padding and stride "2" are used for each convolutional layer to yield the respective output size shown above. In this model, there are total 150,023 trainable parameters. The fully connected layers generally are not required to place after the flatten layer in DCGAN [44].

which the adversarial balance can be enabled to achieve the reliable model performance. In other words, the improvement of DCGAN is accomplished through the combat between the discriminator and generator during training. Either the discriminator or the generator is dominant than the other will result in the early ending of combat, thereby yielding the bad performance of DCGAN. The details of adversarial effect between the discriminator and generator will be discussed later (Section 3.3.5).

To ensure the performance, the architecture of DCGAN needs to be configured through properly tuning a set of hyperparameters. The general guidelines for DCGAN architecture design include [44]: (1) applying LeakyReLU activation functions onto all layers of generator except for the input and output layers; (2) applying tanh activation function onto the output layer of generator; (3) applying LeakyReLU activation functions onto all convolutional layers of discriminator; (4) applying batch normalization behind the layers to increase the training stability; (5) using strided convolutions instead of pooling layer for down-sampling of the discriminator; (6) using fractional-strided convolutions (deconvolution) instead of the unpooling layer for up-sampling of the generator; and (7) no fully connected hidden layers needed that allows deeper architecture. The notable advantage of convolution operations mentioned in (5) and (6) is that the positional connectivity exists between the input values and the output values. Hence, the convolutional and deconvolution layers have parameters to be tuned, while the pooling and unpooling layers don't. Generally, the models are kept with small scale to minimize the chance of overfitting given the relatively small-sized dataset. The appropriate architecture design enables the models without the underfitting and overfitting. For this particular neural network, i.e., DCGAN, the appropriate architecture

Table 2
Generator configuration.

Layer ID	Layer Type	Output Shape	Parameters
#1	Input	100	0
#2	Dense (projection) (LeakyReLU)	8,192	827,392
#3	Reshape	$8 \times 8 \times 128$	
#4	Deconvolution (LeakyReLU)	$16 \times 16 \times 128$	262,272
#5	Deconvolution (LeakyReLU)	$32 \times 32 \times 128$	262,272
#6	Deconvolution (LeakyReLU)	$64 \times 64 \times 128$	262,272
#7	Output (tanh)	$64 \times 64 \times 1$	8,193

Note: The dense layer (#2) is used to create the information required for the synthetic image from latent space. Output layer essentially is a convolutional layer convoluted by a filter with size $8 \times 8 \times 1$ and with the "tanh" activation function. The output layer in the generator should have the same size with the input layer of the discriminator, since the serial connection of discriminator and generator is needed to train the generator. The total number of trainable parameters is 1,622,401.

Table 3
Data specification for fault diagnosis analysis.

Type	Fault condition	Training data size		Testing data size
		Labeled	Unlabeled	
1	Healthy	10	93	11
2	Missing tooth	10	93	11
3	Crack	10	93	11
4	Spalling	10	93	11
5	Chipping_tip_5 (least severe)	10	93	11
6	*Chipping_tip_4*	*0*	*93*	*11*
7	Chipping_tip_3	10	93	11
8	*Chipping_tip_2*	*0*	*93*	*11*
9	Chipping_tip_1 (most severe)	10	93	11
	Total	70	837	99

Note: the data split is randomly implemented upon the specified data size.

design also considers avoiding the model collapse as will be introduced subsequently (Section 3.3.5). Underfitting, overfitting and model collapse can be monitored in terms of the training-validation/testing performance with respect to the training progress. The finalized architecture of DCGAN used in this research and its detailed configurations can be seen in Fig. 6, and Tables 1, 2, respectively.

Unlike the general convolutional neural networks (CNNs) that are constructed upon the down-sampling strategy, the generator model is constructed upon the up-sampling strategy in terms of the random noise input produced from a prespecified latent space. Its key purpose is to learn to map points in the latent space to generated image. The latent space allows one to explore its structure, such as by interpolating between points and performing vector arithmetic between points aiming at achieving targeted effects on the generated image. Generator training requires synergistic interaction between the generator and the discriminator. The training loss of the generator is assessed by the binary classifier's decision on the synthetic image from the generator. The penalty will be applied if the generator fails to deceive the binary classifier.

The binary classifier is expected to identify the authenticity of the image data. The binary classifier training together with the generator training form a closed loop. While the multi-class classifier is only trained upon the small number of labeled data, its parameters are also subject to update during the training of the binary classifier because they are shared with the binary classifier. As a result, the multi-class classifier will affect the generator as well. The unique feature of DCGAN is the encapsulation of those interactive models, which ensures the good semi-supervised learning capability under the limited data with limited known fault labels.

3.3. DCGAN implementation in gearbox diagnosis and case illustration

In this section we present DCGAN implementation details and demonstrate its fault diagnosis performance. The different influential factors with respect to the model performance are systematically investigated and discussed.

3.3.1. Semi-supervised fault classification analysis

Since unlabeled data can be easily acquired from real-time operating machine, the fault conditions contained in the unlabeled data generally are more diverse than that in the labeled data. For example, the chipping tip, which is a continuous type of fault, essentially exhibits continuous fault severities in actual practice. However, the data with limited number of continuous severities can only be acquired experimentally as the labeled data. To mimic that the fault conditions of labeled data are a subset of that of unlabeled data, in this case illustration we purposely remove the data with 2 chipping tip faults and retain the rest with 7 fault conditions as the known/labeled data. This setup can benefit the validation of extended fault diagnosis capability. The data split detail is given in Table 3.

10 labeled data are randomly selected from 93 unlabeled data for each fault condition, yielding totally 70 labeled data for training.

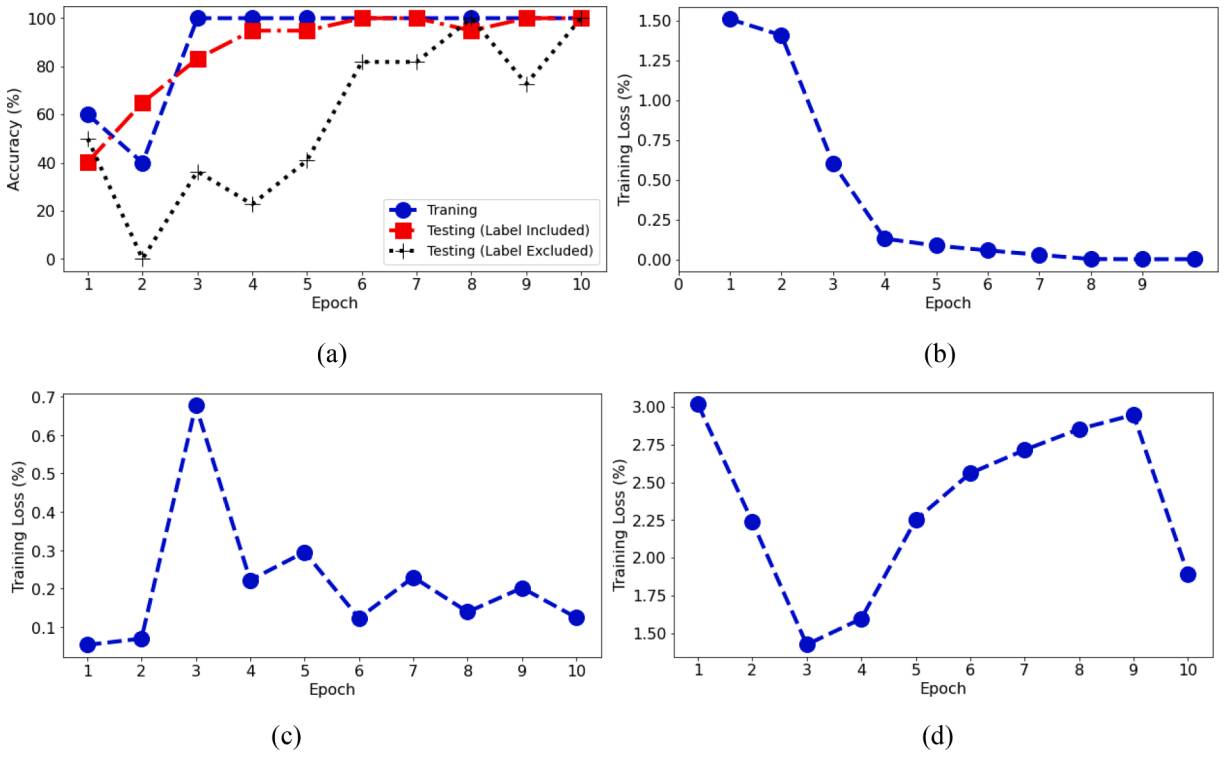

Fig. 7. Model learning and classification performance (a) training and testing accuracy histories of multi-class classifier (discriminator); (b) training loss of multi-class classifier (discriminator); (c) training loss of binary classifier (discriminator); (d) training loss of generator.

The small size of labeled data can substantially reduce the data acquisition cost, and thus highlight the merit of this new method. Total size of available dataset is the sum of unlabeled training and testing data sizes, which is 936 consistent with that mentioned in Section 3.1. Choosing 11 out of 104 samples in each fault condition yields 99 testing data samples, which are statistically independent with respect to the training data samples. Chipping_tip_2 and chipping_tip_4 are particularly hidden in labeled training data to emulate the inclusion relation between the labeled and unlabeled data. In addition to the regular 77 data samples, we also consider 22 unseen data samples of these two chipping chips as testing data since we want to thoroughly investigate the fault diagnosis performance, both the regular and extended fault diagnosis capability. These unseen data can be simply defined as unlabeled testing data. We hope this new method can successfully classify the unseen data as being close to the neighboring fault conditions. For example, If the data of chipping_tip_4, which is unseen/unlabeled by the DCGAN in training, is classified as being close to chipping_tip_3 or chipping_tip_5, then this decision making is considered accurate/reliable. This will be utilized for classification accuracy evaluation of unlabeled/unseen testing data. For illustration purpose, we separately compute the classification accuracy of these two different types of testing data.

Prior to implementing the fault diagnosis practice, the hyperparameters mentioned in Section 2, such as the epoch size, batch size and so on, are investigated to avoid both underfitting and overfitting. In this case, we use stochastic gradient decent algorithm (SGD) for model parameter optimization with the epoch and batch sizes, and learning rate that are finalized as 10, 10 and 0.0002, respectively. The algorithm is developed in Anaconda, Spider IDE using the built-in libraries, i.e., Keras and Tensorflow. With the above-outlined problem setup, we can perform the DCGAN training, in which the training and testing performance with respect to epoch can be monitored shown in Fig. 7. In particular, Fig. 7a illustrates the training and testing accuracy histories of the multi-class classifier. Clearly, both training and testing accuracy will be improved as training proceeds, eventually reaching 100 % accuracy. Neither overfitting nor underfitting is observed, showing the good training process. Two testing accuracy curves are given to demonstrate the classification accuracy of 77 labeled and 22 unlabeled testing samples, respectively. Both testing accuracy yielded is high, which shows that the model can fulfill the actual fault diagnosis mission very well. Both testing accuracy curves exhibit similar increasing tendency. The major difference however lies in that the testing accuracy of labeled data is better than that of unlabeled data during the training. Additionally, the testing accuracy curve of unlabeled data exhibits more oscillation than that of labeled data. Such observations are anticipated as the extended diagnosis of the faults that are beyond the faults in the training data is more challenging because of the prediction uncertainties. Also, the smaller number of unlabeled testing data samples, i.e., 22, may be another influential factor. In contrast to the training accuracy trend (Fig. 7a), the categorical cross-entropy loss of multi-class classifier will gradually decrease along the training (Fig. 7b). Fig. 7a and 7b indicate the good performance of the multi-class classifier built upon semi-supervised learning scheme.

While the inference capability of the multi-class classifier is the major focus in DCGAN model development that has been clearly

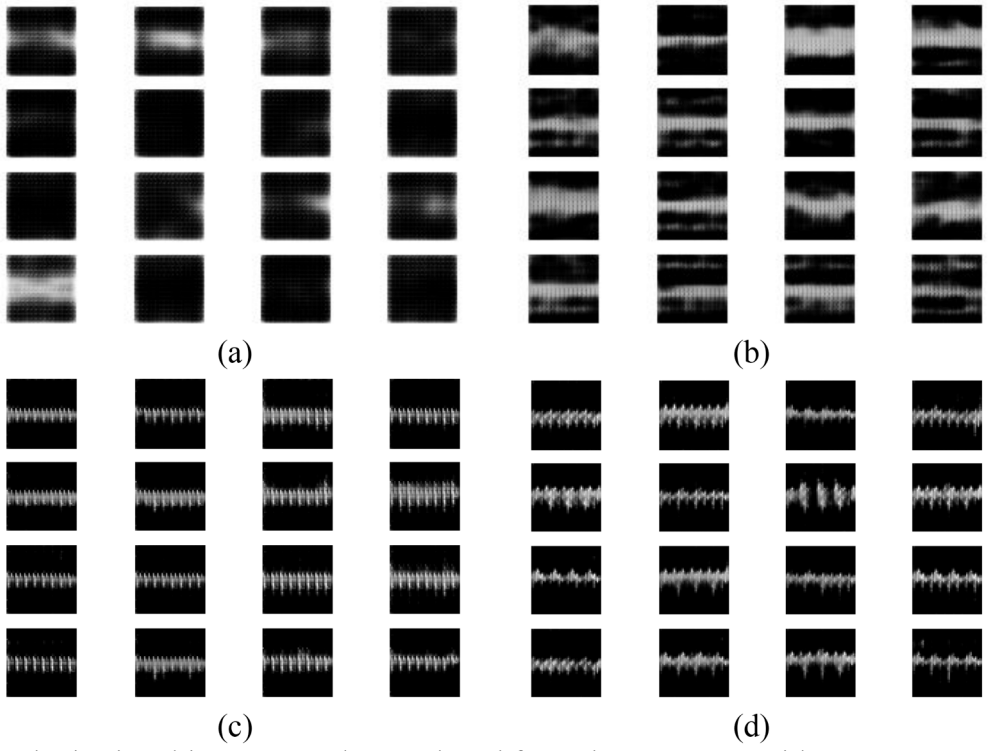

Fig. 8. Synthetic signal image samples produced from the generator with respect to training epoch (each dataset including 16 random samples for illustration) (a) epoch 2; (b) epoch 4; (c) epoch 7; (d) epoch 10 (end of training).

Table 4
Cross-validation results of DCGAN for gear fault diagnosis.

	Run 1	Run 2	Run 3	Run 4	Run 5	Bias	Variance
Training (%)	100	100	100	100	100	0	0
Testing (labels included) (%)	98.70	98.70	100	100	100	0.52	0.51
Testing (labels excluded) (%)	90.91	95.46	100	100	86.37	5.45	35.09

illustrated, it is still worth looking into the training losses of both binary classifier and generator to understand the underlying principle of DCGAN. As can be seen in Fig. 7c and d, the training losses of both generator and binary classifier exhibit the oscillation with respect to epoch, which specifically behave oppositely. This is a clear indication of the balanced combat between the generator and binary classifier during the training, which is fundamentally due to the minmax game theory introduced in Section 2. Through such combat, both training losses eventually will reduce, leading to the enhancement of the multi-class classifier. In other words, the performance of the multi-class classifier (Fig. 7a and b) is highly dependent on the relation between binary classifier and generator along the training (Fig. 7c and d). It is worth pointing out that, the improvement of the generator has the advantage in actual practice especially when the data is scarce. It not only can promote the synergistic enhancement of models in the DCGAN, but also can enrich the dataset for the ease of other supervised learning tasks. In literature [38], the high-fidelity generator was used to produce the plausible synthetic images, which can enrich the image database to facilitate the fault detection. In this research, the synthetic gear vibration signal samples produced via generator with respect to training progress, are demonstrated in Fig. 8. Those synthetic samples fundamentally are transformed from random noise vectors in the latent space. The different random noise vectors will be sampled at different epochs in order to improve the generalization capability of generator to produce various synthetic data to deceive the discriminator. Therefore, the sample images at the same location of sample array in Fig. 8a, b, c and d, are not completely paired. However, it can be seen clearly that the generator will be progressively enhanced to generate the higher-quality sample images as the training proceeds. The sample images after epoch 2 have no agreement with the vibration time-series signals. Conversely, the sample images after epoch 10 (at the end of training) become quite similar with the actual ones (Fig. 5).

3.3.2. Investigation of algorithmic robustness

It is well known that the model construction randomness inevitably exists due to the random training and testing data split, random latent point generation, random batch generation and stochastic optimization. To investigate the model accuracy and robustness from

Table 5
Scenarios created to examine the influence of labeled training data size.

	Scenario 1	Scenario 2	Scenario 3	Scenario 4
Labeled data size	21	35	56	70
Unlabeled data size	837	837	837	837

Note, the testing data size is 99, remaining the same for all scenarios.

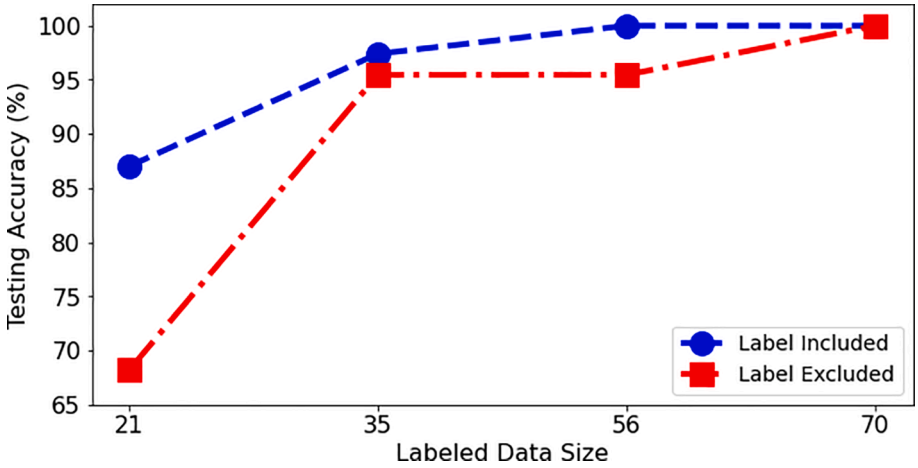

Fig. 9. Testing accuracy with respect to labeled training data size.

the probabilistic perspective, the cross-validation through repeated random sub-sampling is applied [45]. In this statistical validation analysis, 5 folders of data splits are created following the same set-up specified in Section 3.3.1, upon which the corresponding 5 emulation runs are carried out. The final training and testing accuracy obtained after the last epoch in all runs is extracted. We here use the bias-variance decomposition to examine the model performance with the details tabulated in Table 4. The bias and variance are defined respectively as.

$$\text{Bias}(\boldsymbol{\theta}) = E(\boldsymbol{\theta}) - \widehat{\theta} \tag{13a}$$

$$\text{Var}(\boldsymbol{\theta}) = E[(E(\boldsymbol{\theta}) - \boldsymbol{\theta})^2] \tag{13b}$$

where $\widehat{\theta}$ is the nominal accuracy, i.e., 100 %, and $\boldsymbol{\theta}$ is the vector of accuracy values obtained from different emulation runs. Obviously, the training performance is extremely robust and desirable, showing no bias, i.e., 100 % accuracy of all runs. The testing accuracy of both labeled and unlabeled data degrades slightly as compared to the training accuracy, which is a normal phenomenon in model learning process. However, both types have high testing accuracy, indicating the excellent fault diagnosis performance. The testing accuracy of unlabeled data has the higher bias and variance than that of labeled data, which provides the consistent observation as shown in Fig. 7a. Recall the challenge of extended fault diagnosis that has been pointed out. The relatively inferior testing performance for unlabeled data is reasonable. The cross-validation results clearly indicate the desired accuracy and robustness of DCGAN for gear fault diagnosis proposed in this research.

3.3.3. Investigation of effects of labeled and unlabeled data sizes

As highlighted, the strength of semi-supervised learning for gear fault diagnosis is that it can take advantage of a very small number of labeled data combined with a relatively large number of unlabeled data to realize the desired diagnosis performance. Intuitively, either increasing the labeled data or unlabeled data will enhance the model learning adequacy. It however remains unknown what is the best size of labeled or unlabeled size required to ensure the desired performance and meanwhile minimize the data acquisition cost. Because the trade-off between the acquisition cost and model performance naturally exists, it is critical to determine the optimal data size to facilitate the diagnosis analysis. Therefore, here we aim to quantitatively investigate the effects of both labeled data and unlabeled data sizes on the model classification/testing accuracy.

We first carry out the investigation of influence due to labeled data size. For comparison purpose, we create 4 scenarios with different sizes of labeled training data as demonstrated in Table 5. Scenario 4 is the baseline case that has been analyzed in Section 3.3.1. The detailed data split of this scenario can be referred to Table 3. The data splits of other scenarios can be easily identified by adjusting the values in "Labeled" column (Table 3). Following the same analysis procedures in Section 3.3.1, we can collect the testing accuracy information at the end of epoch for all scenarios and obtain the testing accuracy curves with respect to the labeled data size shown in Fig. 9. It is unsurprisingly found that increasing the labeled data size will lead to the accuracy enhancement. Once more than

Table 6
Scenarios created to examine the influence of unlabeled training data size.

	Scenario 1	Scenario 2	Scenario 3	Scenario 4
Unlabeled data size	279	468	648	837
Testing data size	657	468	288	99

Note, the labeled data size is 90, remaining the same for all scenarios. Training data size is the sum of the labeled and unlabeled data sizes.

Table 7
Data split in Scenario 3.

Type	Fault condition	Training data size		Testing data size
		Labeled	Unlabeled	
1	Healthy	10	72	32
2	Missing tooth	10	72	32
3	Crack	10	72	32
4	Spalling	10	72	32
5	Chipping_tip_5 (least severe)	10	72	32
6	*Chipping_tip_4*	*0*	*72*	*32*
7	Chipping_tip_3	10	72	32
8	*Chipping_tip_2*	*0*	*72*	*32*
9	Chipping_tip_1 (most severe)	10	72	32
	Total	70	648	288

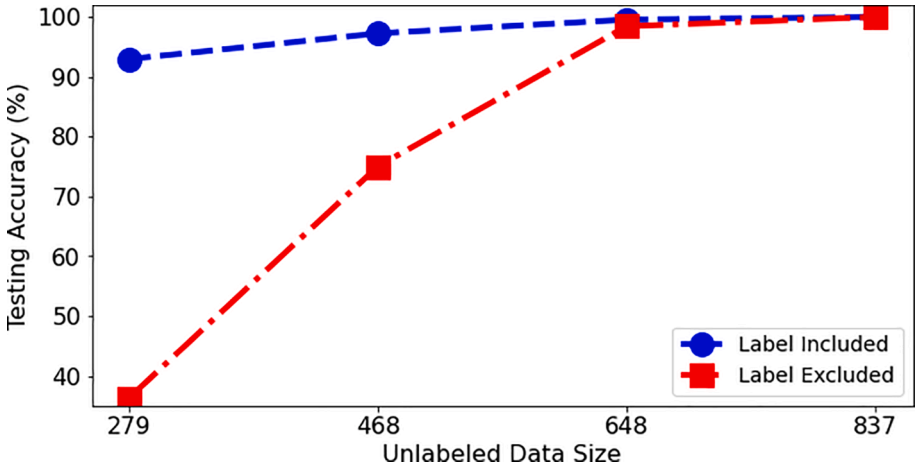

Fig. 10. Testing accuracy with respect to unlabeled training data size.

35 labeled data is incorporated for training, both the testing accuracy of labeled and unlabeled data can be yielded higher than 93 %. On the other hand, both the testing accuracy will have a drastic drop when only using 21 labeled data, especially the testing accuracy of unlabeled data which even decreases to around 67 %. This may indicate the higher sensitivity of extended diagnosis performance with respect to the labeled training data size. Again, the testing labeled data shows the better accuracy than unlabeled testing data under the same labeled training data size.

We then analyze the impact of the unlabeled training data size. For implementation convenience, here we vary the unlabeled data size through adjusting the testing data size since the sum of these two is a constant, i.e., total data size. Similarly, we also create 4 scenarios to correspond to different unlabeled data sizes as shown in Table 6. We specifically give the data split information of scenario 3 (Table 7), which can be compared with that of scenario 4, i.e., baseline case (Table 3) to illustrate the data split difference among scenarios. In all scenarios, the size of labeled training data is kept the same, i.e., 70. The result can be seen in Fig. 10. Obviously, the unlabeled training data size also plays an important role on model performance. Increasing the unlabeled training data size certainly will produce the positive consequence. 648 appears to be a good unlabeled training data size to ensure the desirable testing accuracy of both labeled and unlabeled data. Reducing the unlabeled training data size will more substantially degrade the testing accuracy of unlabeled data than that of labeled data. While the unlabeled training data lack fault labels, they may contain rich pivot features that can be implicitly related to the various actual fault conditions. In other words, this implicit elucidation of physics among different faults can be harnessed by involving the unlabeled data, which thus favors the extended fault diagnosis.

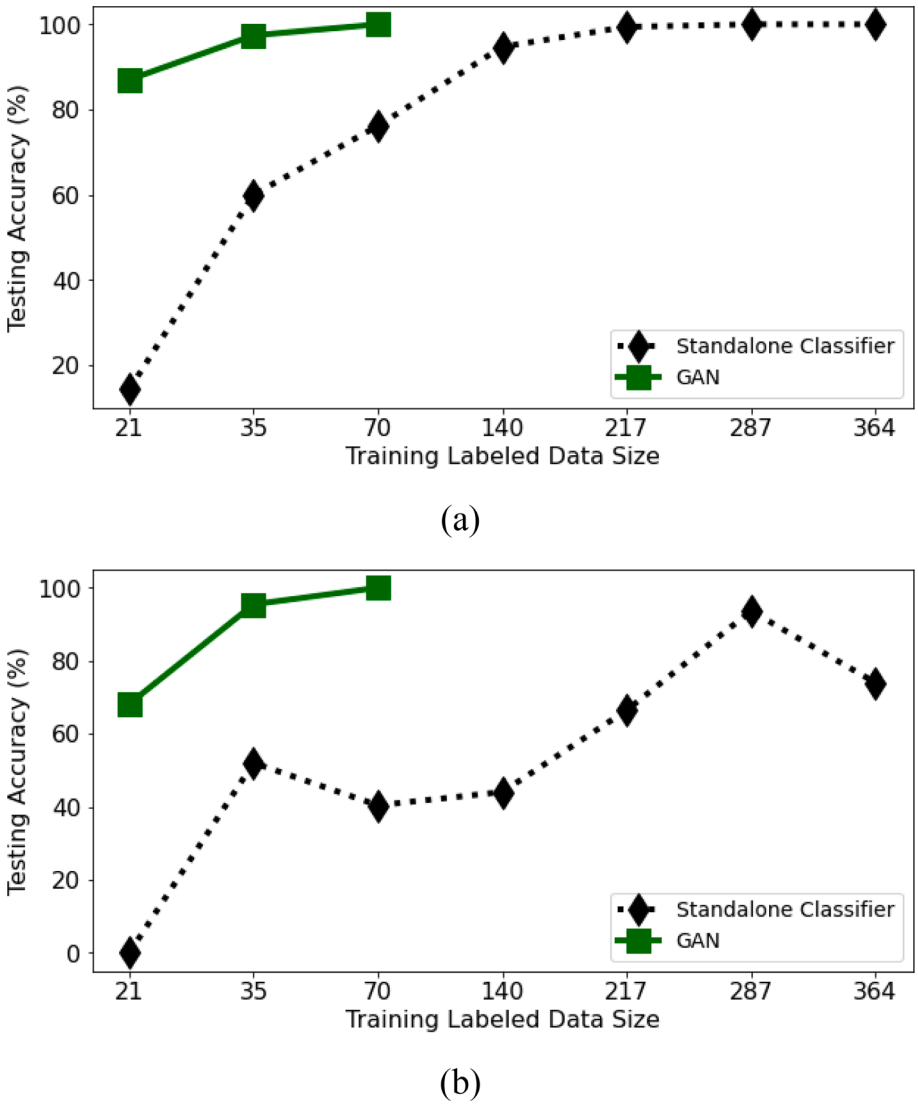

Fig. 11. Testing accuracy comparison between DCGAN and standalone multi-class classifier (a) testing samples with labels included; (b) testing samples with labels excluded.

3.3.4. Performance comparison with respect to standalone multi-class classifier/discriminator

To illustrate how the novel design of DCGAN enhances the inference capability, here we further carry out the comparison with respect to the standalone multi-class classifier that is embedded in DCGAN. The configuration details of this standalone classifier can be referred to as the first 7 layers in Table 1. As a supervised learning model, this standalone multi-class classifier can be independently employed to achieve the fault diagnosis. Because of the supervised learning nature, only the labeled training data size is considered as the influential parameter. Different sizes of the labeled training data thus are used, and the corresponding testing accuracy is evaluated to obtain the performance curves (Fig. 11). The partial result of the DCGAN shown in Fig. 9 is incorporated into Fig. 11 for comparison. It can be observed that at least 217 labeled training data samples are required for the standalone classifier to maintain the desired testing accuracy of labeled data, i.e., higher than 95 %. However, the testing accuracy of unlabeled data stays low regardless of the labeled training data size when using the standalone classifier. It implies that the conventional supervised learning neural network cannot accomplish the extended fault diagnosis. On the other hand, the good performance of DCGAN on the extended fault diagnosis can be realized because the physical correlation between the unseen and known gear faults can be characterized from respective vibration signals via semi-supervised learning upon both the unlabeled and labeled training data. In addition to the performance comparison with the supervised learning technique, investigating other potential semi-supervised learning techniques, such as pseudo labeling [46], semi-supervised transfer learning [47], semi-supervised autoencoder [48] may lead to the thorough validation of the proposed methodology, which will be subject to future research.

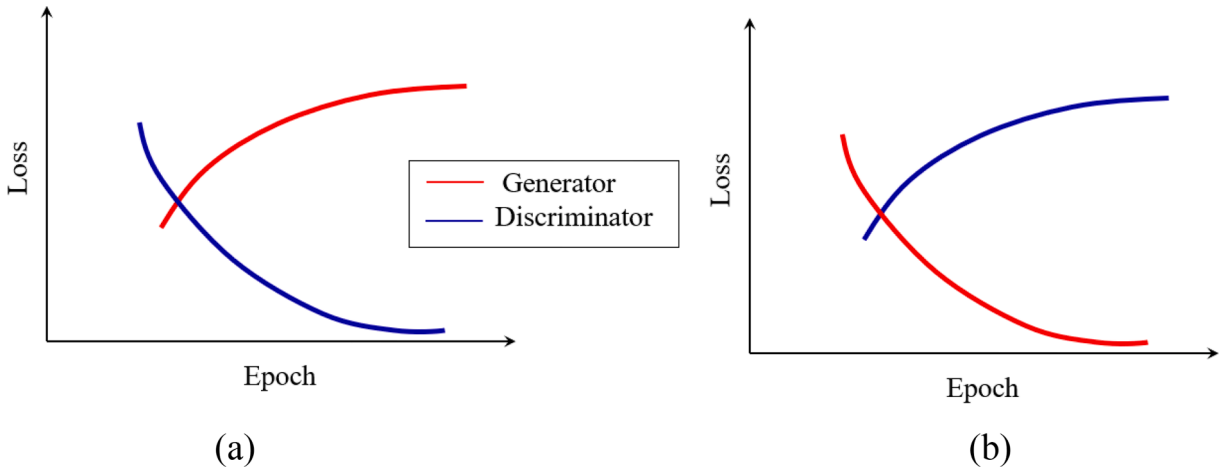

Fig. 12. General GAN training failure modes (a) generator collapse; (b) discriminator collapse.

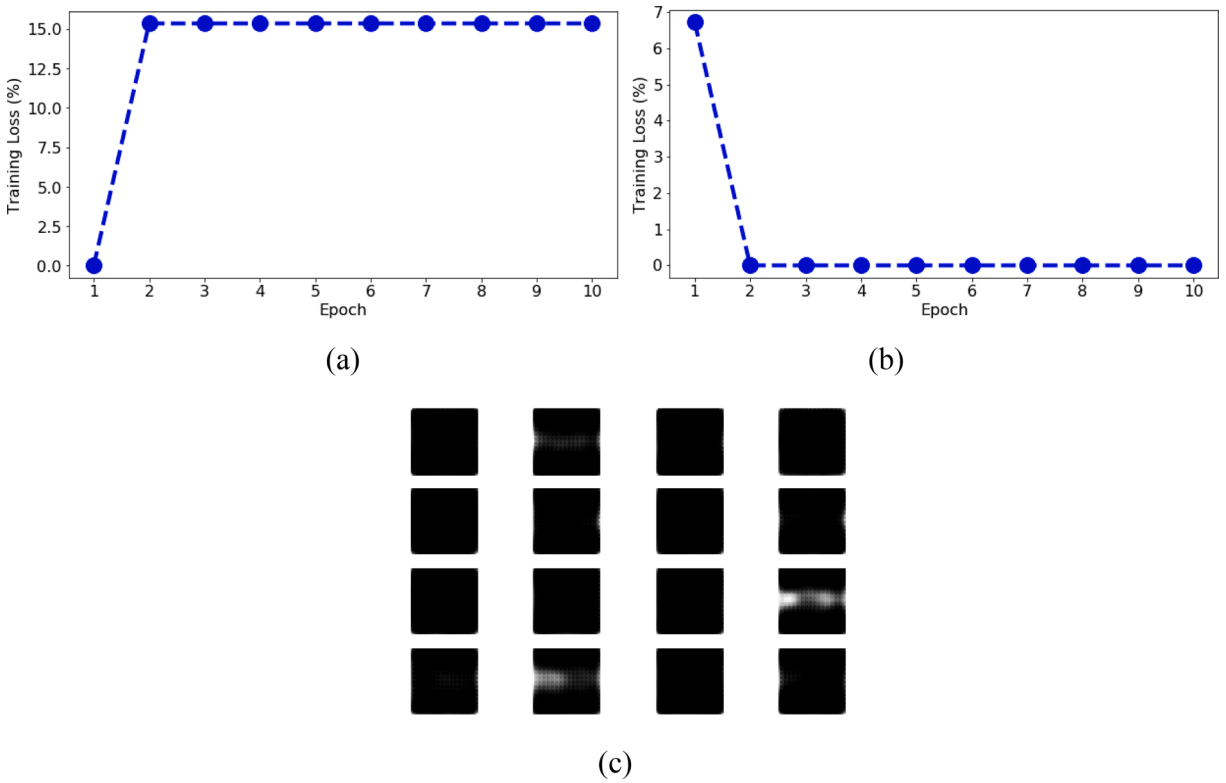

Fig. 13. Illustration of discriminator collapse (a) binary classifier loss history; (b) generator loss history; (c) synthetic signal image samples produced via trained generator.

3.3.5. Discussion of model convergence adversarial effect between discriminator and generator

Because of the adversarial interaction between the discriminator and generator during training, specific care should be taken on the DCGAN architecture configuration to avoid the occurrence of training failure modes, also known as mode collapse. There are two typical failure modes corresponding to two diverged situations either induced by the dominance of the discriminator or the dominance of the generator, which are illustrated in Fig. 12. Specifically, when the discriminator dominates over the generator, the loss of discriminator will be continuously decreased while the loss of generator will be continuously increased. Such trend will become completely opposite if the generator dominates over the discriminator. The model collapse will lead to the ineffective training since the discriminator or generator cannot push the other to continuously improve. On the contrary, the good sign of DCGAN training is that these two types of losses fluctuate and gradually reduce, converging to the steady state.

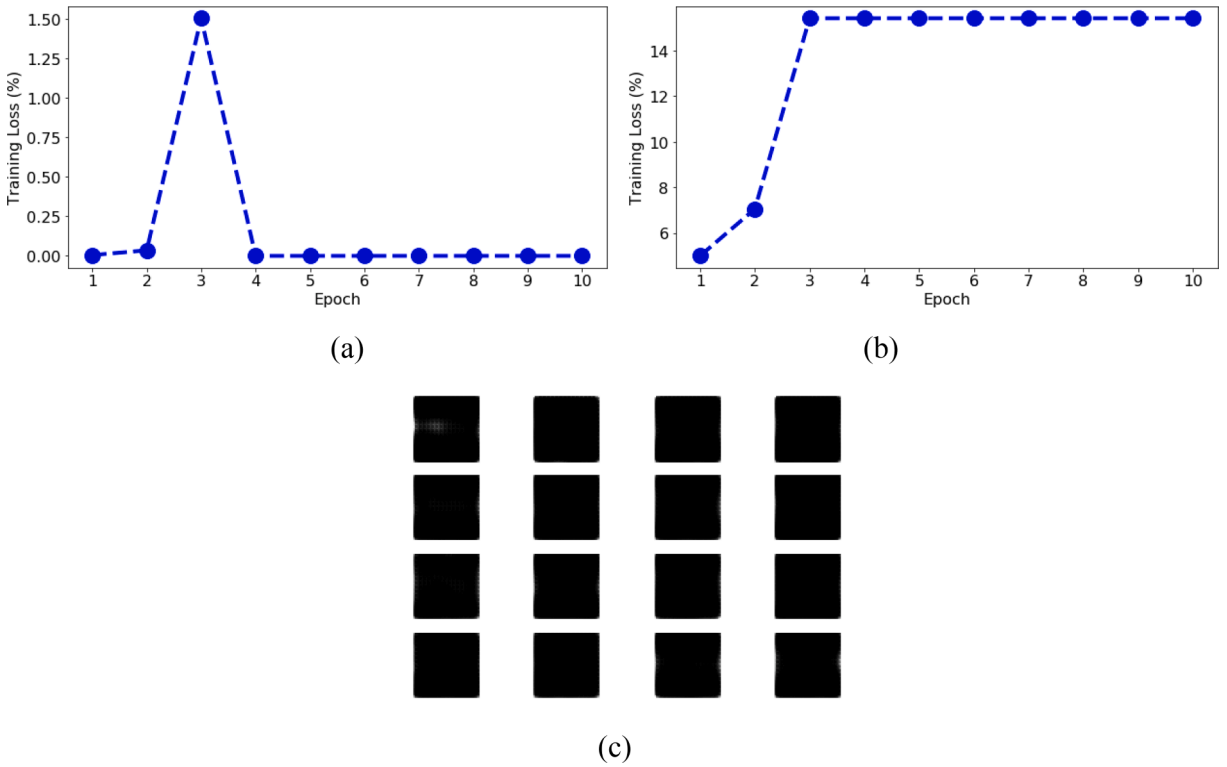

Fig. 14. Illustration of generator collapse (a) binary classifier loss history; (b) generator loss history; (c) synthetic signal image samples produced via trained generator.

In this research, we also examine the model collapse phenomenon. The result in this research (Fig. 7) has demonstrated the proper training of DCGAN for gear fault diagnosis because of no failure modes observed. It is worth pointing out that, the aforementioned failure modes also will be induced if the DCGAN architecture configuration is inappropriately customized. For example, with a simple-structured discriminator (e.g., the small numbers of layers and unknown parameters to be updated), the generator can easily deceive the discriminator, yielding the fairly large discriminator loss and small generator loss at initial stage (Fig. 13a, b). Such adversarial unbalance is difficult to be overturned along the training. It is noticed that the synthetic vibration images (Fig. 13c) produced via generator are notably different from the actual vibration images even though the generator loss is zero. The reason is that the generator loss essentially is measured by the discriminator. The discriminator with ineffective training (large loss observed) cannot provide the reliable assessment of the generator performance. At this point, the generator does not truly deceive the discriminator, and thus the synthetic images generated are implausible. Conversely, the generator collapses which will produce limited varieties of synthetic samples. The discriminator hence becomes too powerful that makes the generator gradient vanish and generator finally will learn nothing. This type of failure mode in this research also can be induced by purposely designing a complex discriminator, as shown in Fig. 14.

In general, the collapse of generator happens (Fig. 14) more frequently than the collapse of discriminator because the generator design is more standard without higher level of flexibility. In order to mitigate such failure mode and meanwhile improve learning stability, some well-established strategies can be adopted, including but not limited to: 1) adding noise into the input image to enhance the generalization of discriminator [49]; 2) using the one-sided smooth labeling [39]; and 3) reducing the complexity of the discriminator as long as it is adequate to learn the data features by increasing the number of epochs.

4. Concluding remarks

In this research, a new semi-supervised method based upon the deep convolutional generative adversarial network (DCGAN) is established to diagnose the gear faults with limitations in training data. The underlying advantage of semi-supervised deep learning is that it allows the combination of both the limited labeled data and large amount of inexpensive unlabeled data for DCGAN training. It is found that the architecture configuration of DCGAN is critically important to ensure the adversarial balance between the generator and discriminator in DCGAN during the training. This will mitigate the model collapse, and hence ensure the desired model performance. Leveraging upon the semi-supervised leaning scheme of DCGAN, the inherent physical correlation between the known and unseen faults can be implicitly explained from the respective vibration signals, which fulfills the extended gear fault diagnosis mission. The fault diagnosis practice has been implemented into the lab-scale gear system using the experimentally measured vibration signals,

where the effectiveness of the proposed method is comprehensively analyzed. This new method shows good accuracy and robustness for both the regular and extended gear fault diagnosis. The influences of parameters on the diagnosis performance also are systematically investigated and reported.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This research is supported in part by NSF under grant CMMI – 2138522, and in part by NSF under grant IIS – 1741174.

References

[1] N. Sawalhi, R.B. Randall, Gear parameter identification in a wind turbine gearbox using vibration signals, Mech. Syst. Signal Process. 42 (2014) 368–376, https://doi.org/10.1016/j.ymssp.2013.08.017.
[2] L.F. Villa, A. Reñones, J.R. Perán, L.J. De Miguel, Statistical fault diagnosis based on vibration analysis for gear test-bench under non-stationary conditions of speed and load, Mech. Syst. Signal Process. 29 (2012) 436–446, https://doi.org/10.1016/j.ymssp.2011.12.013.
[3] Y. Zhang, W. Lu, F. Chu, Planet gear fault localization for wind turbine gearbox using acoustic emission signals, Renew. Energy. 109 (2017) 449–460, https://doi.org/10.1016/j.renene.2017.03.035.
[4] B. Gao, Y. He, W.L. Woo, G.Y. Tian, J. Liu, Y. Hu, Multidimensional tensor-based inductive thermography with multiple physical fields for offshore wind turbine gear inspection, IEEE Trans. Ind. Electron. 63 (2016) 6305–6315, https://doi.org/10.1109/TIE.2016.2574987.
[5] S. Wang, G. Cai, Z. Zhu, W. Huang, X. Zhang, Transient signal analysis based on Levenberg-Marquardt method for fault feature extraction of rotating machines, Mech. Syst. Signal Process. 54 (2015) 16–40, https://doi.org/10.1016/j.ymssp.2014.09.010.
[6] M. Buzzoni, J. Antoni, G. D'Elia, Blind deconvolution based on cyclostationarity maximization and its application to fault identification, J. Sound Vib. 432 (2018) 569–601, https://doi.org/10.1016/j.jsv.2018.06.055.
[7] C. Li, R.V. Sanchez, G. Zurita, M. Cerrada, D. Cabrera, R.E. Vásquez, Gearbox fault diagnosis based on deep random forest fusion of acoustic and vibratory signals, Mech. Syst. Signal Process. 76–77 (2016) 283–293, https://doi.org/10.1016/j.ymssp.2016.02.007.
[8] M. Cerrada, G. Zurita, D. Cabrera, R.V. Sánchez, M. Artés, C. Li, Fault diagnosis in spur gears based on genetic algorithm and random forest, Mech. Syst. Signal Process. 70–71 (2016) 87–103, https://doi.org/10.1016/j.ymssp.2015.08.030.
[9] I. Vamsi, G.R. Sabareesh, P.K. Penumakala, Comparison of condition monitoring techniques in assessing fault severity for a wind turbine gearbox under non-stationary loading, Mech. Syst. Signal Process. 124 (2019) 1–20, https://doi.org/10.1016/j.ymssp.2019.01.038.
[10] G.-D. Sun, Y.-R. Wang, C.-F. Sun, Q. Jin, Intelligent detection of a planetary gearbox composite fault based on adaptive separation and deep learning, Sensors (Switzerland). 19 (2019), https://doi.org/10.3390/s19235222.
[11] W. You, C. Shen, D. Wang, L. Chen, X. Jiang, Z. Zhu, An intelligent deep feature learning method with improved activation functions for machine fault diagnosis, IEEE Access 8 (2020) 1975–1985, https://doi.org/10.1109/ACCESS.2019.2962734.
[12] P. Chen, Y. Li, K. Wang, M.J. Zuo, An automatic speed adaption neural network model for planetary gearbox fault diagnosis, Measurement 171 (2021), 108784, https://doi.org/10.1016/j.measurement.2020.108784.
[13] D. Wang, Y. Chen, C. Shen, J. Zhong, Z. Peng, C. Li, Fully interpretable neural network for locating resonance frequency bands for machine condition monitoring, Mech. Syst. Signal Process. 168 (2022), 108673, https://doi.org/10.1016/j.ymssp.2021.108673.
[14] T. Li, Z. Zhao, C. Sun, L. Cheng, X. Chen, R. Yan, R.X. Gao, WaveletKernelNet: An interpretable deep neural network for industrial intelligent diagnosis, IEEE Trans. Syst. Man, Cybern. Syst. 52 (2021) 2302–2312, https://doi.org/10.1109/TSMC.2020.3048950.
[15] H. Shao, M. Xia, J. Wan, C.W. De Silva, Modified stacked autoencoder using adaptive Morlet wavelet for intelligent fault diagnosis of rotating machinery, IEEE/ASME Trans. Mechatronics. 27 (2022) 24–33, https://doi.org/10.1109/TMECH.2021.3058061.
[16] J. Grezmak, J. Zhang, P. Wang, K.A. Loparo, R.X. Gao, Interpretable convolutional neural network through layer-wise relevance propagation for machine fault diagnosis, IEEE Sens. J. 20 (2020) 3172–3181, https://doi.org/10.1109/JSEN.2019.2958787.
[17] D. Zhang, Y. Chen, F. Guo, H.R. Karimi, H. Dong, Q. Xuan, A new interpretable learning method for fault diagnosis of rolling bearings, IEEE Trans. Instrum. Meas. 70 (2021) 1–10, https://doi.org/10.1109/TIM.2020.3043873.
[18] R. Yan, F. Shen, C. Sun, X. Chen, Knowledge transfer for rotary machine fault diagnosis, IEEE Sens. J. 20 (2020) 8374–8393, https://doi.org/10.1109/JSEN.2019.2949057.
[19] Z. He, H. Shao, P. Wang, J. (Jing) Lin, J. Cheng, Y. Yang, Deep transfer multi-wavelet auto-encoder for intelligent fault diagnosis of gearbox with few target training samples, Knowledge-Based Syst. 191 (2020), 105313, https://doi.org/10.1016/j.knosys.2019.105313.
[20] W. Chen, Y. Qiu, Y. Feng, Y. Li, A. Kusiak, Diagnosis of wind turbine faults with transfer learning algorithms, Renew. Energy. 163 (2021) 2053–2067, https://doi.org/10.1016/j.renene.2020.10.121.
[21] P. Cao, S. Zhang, J. Tang, Preprocessing-free gear fault diagnosis using small datasets with deep convolutional neural network-based transfer learning, IEEE Access 6 (2018) 26241–26253, https://doi.org/10.1109/ACCESS.2018.2837621.
[22] Y. Kim, K. Na, B.D. Youn, A health-adaptive time-scale representation (HTSR) embedded convolutional neural network for gearbox fault diagnostics, Mech. Syst. Signal Process. 167 (2022), 108575, https://doi.org/10.1016/j.ymssp.2021.108575.
[23] Y.A. Yucesan, F.A.C. Viana, A hybrid physics-informed neural network for main bearing fatigue prognosis under grease quality variation, Mech. Syst. Signal Process. 171 (2022), 108875, https://doi.org/10.1016/j.ymssp.2022.108875.
[24] S. Shen, H. Lu, M. Sadoughi, C. Hu, V. Nemani, A. Thelen, K. Webster, M. Darr, J. Sidon, S. Kenny, A physics-informed deep learning approach for bearing fault detection, Eng. Appl. Artif. Intell. 103 (2021), 104295, https://doi.org/10.1016/j.engappai.2021.104295.
[25] M. Sadoughi, C. Hu, Physics-based convolutional neural network for fault diagnosis of rolling element bearings, IEEE Sens. J. 19 (2019) 4181–4192, https://doi.org/10.1109/JSEN.2019.2898634.
[26] S. Shalev-Shwartz, S. Ben-David, Generative Models, in: Underst. Mach. Learn. From Theory to Algorithms, Cambridge University Press, 2014: pp. 295–308. https://doi.org/10.1017/CBO9781107298019.025.
[27] D. Foster, Generative Deep Learning: Teaching Machines to Paint, Compose, and Play, O'Reilly Media, Write, 2019.
[28] H. Yu, J. Tao, C. Qin, M. Liu, D. Xiao, H. Sun, C. Liu, A novel constrained dense convolutional autoencoder and DNN-based semi-supervised method for shield machine tunnel geological formation recognition, Mech. Syst. Signal Process. 165 (2022), 108353, https://doi.org/10.1016/j.ymssp.2021.108353.
[29] R. He, Z. Tian, M.J. Zuo, A semi-supervised GAN method for RUL prediction using failure and suspension histories, Mech. Syst. Signal Process. 168 (2022), 108657, https://doi.org/10.1016/j.ymssp.2021.108657.
[30] M. Mardani, E. Gong, J.Y. Cheng, S.S. Vasanawala, G. Zaharchuk, L. Xing, J.M. Pauly, Deep generative adversarial neural networks for compressive sensing MRI, IEEE Trans. Med. Imaging 38 (2019) 167–179, https://doi.org/10.1109/TMI.2018.2858752.

[31] K. Zhou, J. Tang, Harnessing fuzzy neural network for gear fault diagnosis with limited data labels, Int. J. Adv. Manuf. Technol. (2021), https://doi.org/10.1007/s00170-021-07253-6.
[32] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial networks, Commun. ACM 63 (2020) 139–144, https://doi.org/10.1145/3422622.
[33] M.R. Pavan Kumar, P. Jayagopal, Generative adversarial networks: a survey on applications and challenges, Int. J. Multimed. Inf. Retr. 10 (2021) 1–24, https://doi.org/10.1007/s13735-020-00196-w.
[34] Z. Wang, Y. Lin, K.-T. (Tim) Cheng, X. Yang, Semi-supervised mp-MRI data synthesis with StitchLayer and auxiliary distance maximization, Med. Image Anal. 59 (2020), 101565, https://doi.org/10.1016/j.media.2019.101565.
[35] L. Mosser, O. Dubrule, M.J. Blunt, Reconstruction of three-dimensional porous media using generative adversarial neural networks, Phys. Rev. E 96 (2017), 043309, https://doi.org/10.1103/PhysRevE.96.043309.
[36] S. Shao, P. Wang, R. Yan, Generative adversarial networks for data augmentation in machine fault diagnosis, Comput. Ind. 106 (2019) 85–93, https://doi.org/10.1016/j.compind.2019.01.001.
[37] Y. Li, W. Zou, L. Jiang, Fault diagnosis of rotating machinery based on combination of Wasserstein generative adversarial networks and long short term memory fully convolutional network, Measurement 191 (2022), 110826, https://doi.org/10.1016/j.measurement.2022.110826.
[38] P. Liang, C. Deng, J. Wu, Z. Yang, Intelligent fault diagnosis of rotating machinery via wavelet transform, generative adversarial nets and convolutional neural network, Meas. J. Int. Meas. Confed. 159 (2020), 107768, https://doi.org/10.1016/j.measurement.2020.107768.
[39] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, X. Chen, Improved techniques for training GANs, Adv. Neural Inf. Process. Syst. (2016) 2234–2242.
[40] N. Souly, C. Spampinato, M. Shah, Semi supervised semantic segmentation using generative adversarial network, in: 2017 IEEE Int. Conf. Comput. Vis., 2017: pp. 5689–5697. https://doi.org/10.1109/ICCV.2017.606.
[41] Z. Zhang, M.R. Sabuncu, Generalized cross entropy loss for training deep neural networks with noisy labels, ArXiv (2018).
[42] S. Zhang, J. Tang, Integrating angle-frequency domain synchronous averaging technique with feature extraction for gear fault diagnosis, Mech. Syst. Signal Process. 99 (2018) 711–729, https://doi.org/10.1016/j.ymssp.2017.07.001.
[43] K. Zhou, J. Tang, Gear dataset, (2022). https://doi.org/10.17632/87y47nvsf4.1.
[44] A. Radford, L. Metz, S. Chintala, Unsupervised representation learning with deep convolutional generative adversarial networks, 4th Int. Conf. Learn. Represent. ICLR 2016 - Conf. Track Proc. (2016) 1–16.
[45] D. Berrar, Cross-validation, Encycl. Bioinforma. Comput. Biol. ABC Bioinforma. 1–3 (2018) 542–545, https://doi.org/10.1016/B978-0-12-809633-8.20349-X.
[46] X. Tao, C. Ren, Q. Li, W. Guo, R. Liu, Q. He, J. Zou, Bearing defect diagnosis based on semi-supervised kernel Local Fisher Discriminant Analysis using pseudo labels, ISA Trans. 110 (2021) 394–412, https://doi.org/10.1016/j.isatra.2020.10.033.
[47] T. Berghout, L.-H. Mouss, T. Bentrcia, M. Benbouzid, A semi-supervised deep transfer learning approach for rolling-element bearing remaining useful life prediction, IEEE Trans. Energy Convers. (2021) 1, https://doi.org/10.1109/TEC.2021.3116423.
[48] X. Wu, Y. Zhang, C. Cheng, Z. Peng, A hybrid classification autoencoder for semi-supervised fault diagnosis in rotating machinery, Mech. Syst. Signal Process. 149 (2021), 107327, https://doi.org/10.1016/j.ymssp.2020.107327.
[49] S. Jenni, P. Favaro, On stabilizing generative adversarial training with noise, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis Pattern Recogn., 2019, pp. 12137–12145, https://doi.org/10.1109/CVPR.2019.01242.