

# **Review** Article

# A Multi-Index Generative Adversarial Network for Tool Wear Detection with Imbalanced Data

Guokai Zhang ,<sup>1</sup> Haoping Xiao,<sup>2</sup> Jingwen Jiang ,<sup>3</sup> Qinyuan Liu ,<sup>2</sup> Yimo Liu ,<sup>4</sup> and Liying Wang <sup>5</sup>

<sup>1</sup>School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai, China
 <sup>2</sup>Department of Computer Science and Technology, Tongji University, Shanghai, China
 <sup>3</sup>Technology Transformation Center, Shanghai Electric Group Co., Ltd. Central Academe, Shanghai, China
 <sup>4</sup>School of Software, Tongji University, Shanghai, China
 <sup>5</sup>Shanghai Leosue Network Technology Co., Ltd., Shanghai, China

Correspondence should be addressed to Qinyuan Liu; liuqy@tongji.edu.cn

Received 21 June 2020; Revised 30 October 2020; Accepted 20 November 2020; Published 7 December 2020

Academic Editor: Min Xia

Copyright © 2020 Guokai Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The scarcity of abnormal data leads to imbalanced data in the field of monitoring tool wear conditions. In this paper, a novel multiindex generative adversarial network (MI-GAN) is proposed to detect the tool wear conditions subject to imbalanced signal data. First, the generator in the MI-GAN is trained to produce fake normal signals, and the discriminator computes scores of testing signals and generated signals. Next, the generator detects abnormal signals based on the performance of imitating testing signals, and the discriminator will compute the scores of testing signals and generated signals. Subsequently, two indexes, i.e.,  $L_2$ -norm and temporal correlation coefficient (CORT), are put forward to measure the similarity between generated signals and testing signals. Finally, our decision-making function further combines  $L_2$ -norm and CORT with two discriminator scores to determine the tool conditions. Experimental results show that our method obtains 97% accuracy in tool wear detection based on imbalanced data without manual feature extraction, which outperforms traditional machine learning methods.

# 1. Introduction

The past decades have witnessed the rapid development of equipment-manufacturing technologies. With the widespread applications of computerized numerical control (CNC) machines, the productivity and accuracy in the manufacturing processes have been significantly enhanced. As an important part of CNC machines, the conditions of the cutting tools are critical to the quality of products. However, since the cutting tools are usually subject to extremely severe rubbing environment, tool wear is inevitable in the manufacturing processes. Due to the frequent friction and extrusion, the cutting tools gradually wear out, lose sharpness, and become dull. The wear of cutting tools might increase the cutting forces and temperature, degrade the machine surface texture, and even lead to fatal failures to normal operation of the manufacturing systems in some extreme cases. Therefore, designing effective real-time tool wear monitoring techniques that can guarantee high quality in the manufacturing process has received much attention from both academia and industry [1-3].

Many attempts have been made to detect tool wear in recent years, among which image-based methods are the most widely used approaches. For example, the authors in [4–6] have developed an image-based online tool condition detection technique by using optical methods. However, it should be noted that such an image-based method is usually vulnerable to the manufacturing environment as the cutting fluid, chips, and dirt might hinder the image acquisition of the workpieces, and hence, the tool conditions can only be measured at the end of one batch. To handle such an issue, some online tool condition detection methods are proposed based on signal analysis. As is well recognized, there exists a certain relationship between the tool wear condition and the manufacturing signals, e.g., vibration, acoustic, and current signals. These signals would present different features when the cutting tool works in different conditions. By deploying specific sensors, manufacturing signals can be easily collected and then exploited to determine the tool wear condition in a real-time manner.

Up to now, various methods have been proposed on tool wear monitoring techniques based on signal analysis. To be specific, the authors in [7] have utilized feedback current signals to classify the tool condition. The performance and accuracy of several pattern recognition techniques, including support vector machine (SVM), linear discriminant analysis (LDA), K-nearest neighbor (KNN), neural network (NN) with one hidden layer, naïve Bayes (NB), and decision trees (DT), have been compared. It has been concluded that, among these classification techniques, LDA and SVM are the most effective ones. Moreover, a comparative study has been conducted in [8] which demonstrates that the predictive capability of random forests (RFs) outperforms artificial neural networks (ANNs) with one hidden layer and support vector regression (SVR). However, all these traditional techniques [7-13] highly rely on the features manually extracted from raw data, which need to analyze characteristics of the manufacturing signals and extract reliable features based on prior knowledge of specific tasks, and thus are cumbersome. Consequently, it is of practical significance to develop an easy-to-use method without the feature extraction procedure.

It should be pointed out that deep learning is a promising technique capable of automatically searching information features behind the raw data. In [14-17], the feasibility of convolutional neural networks (CNNs) on extracting information features has been presented. In fact, one can convert input signals into images using Gramian Angular Summation Fields and then leverage CNN to classify the tool wear condition [18]. In [19], a stacked sparse autoencoder neural network has been presented to diagnose the tool wear conditions. By decomposing original 1D signals to reconstruct 2D ones, CNNs can accurately predict the tool wear condition [20]. However, there are some critical shortcomings of the CNN-based tool condition detection [14-17]. For example, a large labeled dataset is required to improve the performance of the CNN, and the signal-to-image conversion procedure might result in a great information loss of the raw signals [18]. In addition, for the majorities of the industrial processes, the normal and abnormal labels of the data are imbalanced. The low failure rate of the manufacturing machines leads to the scarcity of abnormal data, and with limited abnormal data, it is almost impossible to describe the accurate characteristics of all fault types.

Very recently, generative adversarial networks have been utilized to solve imbalanced data problems in unsupervised anomaly detection. To be specific, the authors in [21] have put forward a generative adversarial network with an encoder-decoder-encoder three-subnetwork generator to handle imbalanced industrial time series, but an obvious shortcoming lies in that it still has to manually extract features. Moreover, some other GAN-based methods such as AnoGAN and GAN-AD have also been proposed to achieve detection targets [22, 23]. In these previous works, an anomaly score is established to detect the abnormal signals by directly combining the residual loss and the discriminator loss. However, it is noted that the magnitude of the residual loss is usually different from that of the discriminator loss, and thus, a direct combination of these two losses would not lead to a satisfactory detection.

Motivated by the above discussion, in this paper, we propose a novel multi-index GAN (MI-GAN) method for tool wear detection involving imbalanced data. To deal with the challenges stemming from data imbalance, the generator of the proposed MI-GAN focuses on learning the characteristics of the normal signals, and the discriminator outputs the scores which represent the fake level of the input signals. Such a network architecture can effectively detect the tool condition by learning how to generate and distinguish normal signals. Especially, a novel multi-index decisionmaking function which incorporates  $L_2$ -norm, temporal correlation coefficient (CORT) [24], the score of the input signal, and the score of the generated signal is developed to aggregate more representative criteria. In the end, we use the temporal convolution networks (TCNs) as the backbone of our generator and discriminator networks to encode more temporal and sequential features, which could further leverage the performance of the model instead of the empirical feature definition. We sample real current signals from a CNC machine in different manufacturing processes and then conduct the tool wear detection experiments. The experimental results demonstrate that our designed MI-GAN could achieve competitive performance on the accuracy, precision, recall, and F-measure in detecting the abnormal tool conditions. Overall, the main contributions of this paper can be highlighted as follows: (1) a novel MI-GAN which can generate and distinguish normal signals is established to solve imbalanced data in monitoring tool wear conditions; (2) a decision-making function with multi-index is developed to aggregate more representative criteria; and (3) the feature extraction ability is enhanced by applying the TCN as the backbone of our MI-GAN generator and discriminator networks without predefined feature selection.

The rest of the paper is organized as follows. Section 2 introduces the methodology of the proposed MI-GAN including the data acquisition procedure, the architecture of the MI-GAN, the training process, and the multi-index decision-making algorithm. In Section 3, we carry out the experimental results to evaluate the performance of our method. Section 4 finally draws conclusions.

#### 2. Structure of Monitoring Systems

In the manufacturing processes of CNC machines, it is more suitable to monitor the state of the tool by sampling the current signals of the spindle motor and analyzing their realtime characteristics. As shown in Figure 1, the spindle transmission system consists of a motor, spindle, timing belt, drive, etc. The main shaft is connected to the motor through a belt, the machine tool sends a control signal to the drive which outputs the spindle motor current to drive the motor



FIGURE 1: The overall structure of the spindle transmission system of the CNC machine tool.

to rotate, and the spindle is derived by the motor via a timing belt. Since the spindle motor current is positively related to the cutting force, the change of the real-time spindle current could reflect the change of the cutting force. With the gradual wear of the tool during the machining process, the cutting performance of the tool will gradually decrease and thus requires greater cutting force which leads to the increase of motor current. Therefore, the current signal of the spindle is utilized to detect the tool conditions throughout the paper.

#### 3. Methodology

A detailed introduction of the proposed MI-GAN will be presented in this section. Generally speaking, the procedure of tool condition detection based on MI-GAN can be roughly divided into the following two stages.

In stage 1, the normal signals are exploited to train the generative adversarial network. The generator tries to synthesize normal signals by minimizing generator loss. Generated signals and normal signals are fed into the discriminator in sequence to train the network. When the training losses converge, the generator can stably generate plausible normal signals, and the discriminator can measure the fake-level scores of actual signals and generated signals. Here, actual signals denote the signals in the dataset.

In stage 2, we will make full use of our pretrained model to calculate four specific indexes (i.e., L2-norm, CORT, and two fake-level scores, which will be further clarified in Section 3.4). First, by inverting input signals to specific noises z, the pretrained generator is able to generate signals similar to input signals. Next, the values of indexes  $L_2$ -norm and CORT can be calculated between the generated signals and the input signals. The fake-level scores of the generated signals and the input signals will be output from the pretrained discriminator. Furthermore, the mean and standard deviation of these four indexes are calculated to determine the detecting thresholds. Eventually, the model compares four indexes of the testing signal with the thresholds to determine either normal signal or abnormal. A brief overview is provided to help understand our method in Figure 2, and all the technical details will be explained in the following sections.

3.1. Data Acquisition. The dataset used is the spindle current signals collected by the CNC machine tool when drilling 3 holes in a workpiece. The sampling frequency of the current signals is 10 Hz, and each 130 sampling point data collected is recorded as a set of signals. In each manufacturing process, the CNC machine tools are monitored in time, and their tool condition (i.e., normal or abnormal) was recorded as a label attached with each set. The dataset can be denoted as  $C = \{C_1, C_2, \ldots, C_N\}$ , where  $C_i = \{X_{i1}, X_{i2}, \ldots, X_{in}\}$  is time series data with the total length *n*, and  $X_{ik}$  is the current value of  $C_i$  at sampling instant *k*. We divide our dataset into two subsets, i.e., training dataset and testing dataset. The training dataset contains 287 normal signals and 11 abnormal signals.

3.2. Architecture of the MI-GAN. The schematic diagram of the MI-GAN architecture is illustrated in Figure 3, which consists of a generator and a discriminator. The generator projects the random noise space to the normal signal space so as to confuse the discriminator. It takes a vector of 130 random numbers independently drawn from a uniform distribution as the input and outputs a fake normal signal of length 130. Both generated signals and normal signals are fed into the discriminator to obtain fake-level scores, which indicate how fake the input signals are. A higher fake-level score indicates that the input signal is more likely to be an abnormal one, and vice versa. Notice that the TCN residual block (as depicted in Figure 4) has excellent sequence modeling performance [25]. Consequently, the generator first utilizes six TCN residual blocks to extract the features of input noise vector z and then a linear layer to synthesize the plausible signal. Similarly, the TCN residual blocks are also exploited as a part of our discriminator, and two linear layers are added to compute fake-level scores.

3.3. Training Generative Adversarial Networks. The classical training method for the generative adversarial networks is to optimize Jensen–Shannon (JS) or Kullback–Leibler divergence between the model distribution and real-data distribution. However, it is pointed out in [26] that such methods are unreliable and might lead to training instability and mode collapse. To avoid this problem, we use Wasserstein divergence [27] as our min-max objective as follows:

$$\min_{G} \max_{D} \mathbb{B}_{G(z) \sim \mathbb{P}_{g}}[D(G(z))] - \mathbb{B}_{C \sim \mathbb{P}_{r}}[D(C)] - 2\mathbb{B}_{\widetilde{C} \sim \mathbb{P}_{u}}\left[\left\|\nabla_{\widetilde{C}}D(\widetilde{C})\right\|^{6}\right],$$
(1)

where  $\mathbb{P}_g$  is the distribution of generated signals,  $\mathbb{P}_r$  is the distribution of real signals,  $\mathbb{P}_u$  is a Radon probability measure, *z* represents random noises, *C* indicates the normal signals, and  $\tilde{C}$  is a sequence sampled from  $\mathbb{P}_u$ . *G*(.) and *D*(.) represent the generator and discriminator, respectively.

Hence, the generative adversarial networks are trained by optimizing the generator loss  $L_G$  and the discriminator loss  $L_D$  given in (2) and (3) as follows:



FIGURE 2: The overall framework of the MI-GAN. In stage 1, the generative adversarial network is trained. Stage 2 makes full use of the pretrained model to calculate four specific indexes.



FIGURE 3: The schematic diagram of the MI-GAN architecture. The generator projects the random noise space to the normal signal space, and the discriminator calculates the final fake-level scores of the abnormal signals.



FIGURE 4: The architecture of TCN residual blocks.

$$L_G = \mathbb{B}_{G(z) \sim \mathbb{P}_q}[D(G(z))], \tag{2}$$

$$L_{D} = \mathbb{B}_{C \sim \mathbb{P}_{g}}[D(C)] - \mathbb{B}_{G(z) \sim \mathbb{P}_{r}}[D(G(z))] + 2\mathbb{B}_{\widetilde{C} \sim \mathbb{P}_{u}}\left[\left\|\nabla_{\widetilde{C}}D(\widetilde{C})\right\|^{6}\right],$$
(3)

where  $\tilde{C} = (1 - \mu)C + \mu G(z)$  and  $\mu$  is a nonnegative scalar belonging to [0, 1].

Inspired by [21, 23], only the normal signals are input into the discriminator for the sake of encouraging the MI-GAN to generate fake normal signals. The generator produces fake normal signals to confuse the discriminator by minimizing  $L_G$ , and the discriminator is trained to compute the fake-level score to distinguish normal signals and fake normal signals by minimizing  $L_D$ . Furthermore, the Adam algorithm [28] is adopted to optimize the generator and discriminator losses. In the implementation, we choose the same weights for *C* and G(z) to compute  $\tilde{C}$ , i.e.,  $\mu = 0.5$ .

3.4. Threshold Determination. We utilized both generator and discriminator to monitor tool wear conditions. To make full use of the generator, the input signals will be projected to the latent space using our pretrained generator. Referring to the inversion technique in [29], we apply Algorithm 1 in Table 1 to search specific noises z as the input of the generator, whose corresponding generated signals  $\tilde{C}$  are similar to the input signals C. This inversion technique uses the RMSprop optimization algorithm [30] to minimize the mean square error of generated signals and input signals.

Since the generator captures only the normal pattern, it is difficult to output fake abnormal signals. When the input signals are abnormal, the generated signals will be dissimilar **Require:** batch size *m*, input signals  $C = (C_1, C_2, ..., C_m)$ , pretrained generator *G*, training iterations *n*, and learning rate  $\alpha$ . (1) Sample noises  $z = (z_1, z_2, ..., z_m)$  from standard normal distribution.

- (2) **for**  $\bar{k} = 1$  to *t* do:
- (3) Generate a batch of fake normal signals:  $\tilde{C} = G(z)$ .
- (4) Calculate the mean square error between C and  $\tilde{C}$  as  $L \leftarrow 1/m \sum_{i=1}^{m} \|C_i \hat{C}_i\|^2$ , where  $\tilde{C}_i = G(z_i)$  is the *i*th element in  $\hat{C}$ .
- (5)  $z \leftarrow \text{RMS prop}(L, \alpha)$ .
- (6) end for
- (7) **return** the specific noises z.

with input signals. Therefore, the generated signals will fall into two categories: one is similar signals whose inputs are normal signals, and the other one is dissimilar signals whose inputs are abnormal signals.

Denote  $\widehat{C}_i = (\widehat{X}_{i1}, \widehat{X}_{i2}, \dots, \widehat{X}_{in}), \quad X_k = (X_{1k}, X_{2k}, \dots, X_{mk}),$  and  $\widehat{X}_k = (\widehat{X}_{1k}, \widehat{X}_{2k}, \dots, \widehat{X}_{mk}).$  In order to measure the similarity between generated signals and input signals, we introduce  $L_2$ -norm (i.e.,  $L(C, \widehat{C})$ ) and first-order CORT (i.e., CORT  $(C, \widehat{C})$ ) as follows:

$$L(C, \widehat{C}) = \sqrt{\sum_{k=1}^{n} \|X_k - \widehat{X}_k\|^2},$$
(4)

$$\operatorname{CORT}(C,\widehat{C}) = \frac{\sum_{k=1}^{n-1} (X_{k+1} - X_k) (\widehat{X}_{k+1} - \widehat{X}_k)^T}{\sqrt{\sum_{k=1}^{n-1} \|X_{k+1} - X_k\|^2} \sqrt{\sum_{k=1}^{n-1} \|\widehat{X}_{k+1} - \widehat{X}_k\|^2}}.$$
(5)

In fact,  $L_2$ -norm in (4) indicates the distance between generated signals and input signals, and the first-order CORT in (5) evaluates their growth behavior similarity. A larger  $L_2$ -norm or a smaller CORT means a larger dissimilarity between generated signals and input signals. That is, the input signals are more probable to be abnormal.

Furthermore, we also utilize our pretrained discriminator to help distinguish whether the input signal is normal. The output of our discriminator can be regarded as a fakelevel score. Since we only feed normal data in the training stage, a higher fake-level score represents the signal is less likely to be a normal one. Therefore, the discriminator will output a high fake-level score for abnormal signals. The fakelevel scores of normal signals will be lower than those of abnormal signals, and their corresponding generated signals will have the same situation. By now, we have put forward four indexes to help identify the tool wear condition, which are  $L_2$ -norm, CORT, the fake-level scores of input signals, and the fake-level scores of generated signals.

Given these indexes, four thresholds could be established to distinguish the normal and abnormal signals. In order to determine the threshold values, we need to calculate four indexes of the normal signals in the training dataset. Subsequently, the mean and standard deviation of the values of the indexes are computed, which are critical for identifying abnormal inputs. Since CORT is inversely related to the possibility of signals to be abnormal and the other three indexes are positively correlated, the thresholds of  $L_2$ -norm of normal signals and two fake-level scores are defined as their mean plus standard deviation, whereas the threshold of CORT is denoted as its mean value minus standard deviation. These thresholds are based on statistic principles and thus can classify the input signals effectively.

3.5. Multi-Index Decision-Making Function. We develop a decision-making function to reach the detection target. For each input signal, the function uses the pretrained model and equations (4) and (5) to obtain four indexes. At first, these indexes are uniformed by subtracting their corresponding thresholds to determine whether the signal is abnormal or not. For uniformed  $L_2$ -norm and two fake-level scores, positive values indicate that the inputs are more likely to be abnormal, and their magnitude describes the corresponding possibility. For uniformed CORT, negative values indicate a higher probability of input signals to be abnormal. In other words, the less similar the generated signals are, the smaller the CORT is. After the uniformed procedure, these indexes are further scaled into the same range by dividing their standard deviation, respectively. Finally, we integrate all the scaled indexes to a comprehensive score, based on which one can infer the tool wear condition. To sum up, our multi-index decision-making algorithm is elaborated in Table 2.

#### 4. Experimental Results

4.1. Evaluation Metrics. As the current signal data are imbalanced, we use accuracy, weighted average recall (*W*-recall), precision, and *F*-measure to evaluate the overall performance of our detection method. Denote TP, FP, TN, and FN as true positive, false positive, true negative, and false negative, respectively. For each class, the recall, precision, and *F*-measure can be calculated as follows:

$$Recall = \frac{TP}{TP + FN},$$

$$Precision = \frac{TP}{TP + FP},$$

$$F - measure = \frac{2 \times Precision \times Recall}{Precision + Recall}.$$
(6)

Moreover, the *W*-recall, *W*-precision, and *W*-*F* metrics are defined as follows:

Algorithm 2

**Require:** mean of  $L_2$ -norm, CORT, fake-level score of normal signals, and fake-level score of the generated signal, denoted as  $L_m$ , CORT<sub>m</sub>,  $F_m(n)$ , and  $F_m(g)$ . Standard deviation of  $L_2$ -norm, CORT, fake-level score of normal signals, and fake-level score of the generated signal, denoted as  $L_{\text{std}}$ , CORT<sub>std</sub>,  $F_{\text{std}}(n)$ , and  $F_{\text{std}}(g)$ . Input signals *C*, pretrained Generator *G*, and pretrained Generator *D*.

- (1) Invert C to the latent space to search noises z using Algorithm 1.
- (2) Generate a batch of signals:  $\hat{C} = G(z)$ .
- (3) Calculate the  $L_2$ -norm and CORT between C and  $\hat{C}$  based on (4) and (5).
- (4) Compute the fake-level scores of C and  $\hat{C}$ , i.e., D(C) and  $D(\hat{C})$ , based on the discriminator
- (5) Subtract thresholds:

(6)  $Inde x_1 = L(C, \widehat{C}) - (L_m + L_{std})$ (7)  $Inde x_2 = CORT(C, \widehat{C}) - (CORT_m - CORT_{std})$ (8)  $Inde x_3 = D(C) - (F_m(n) + F_{std}(n))$ (9)  $Inde x_4 = D(\widehat{C}) - (F_m(g) + F_{std}(g))$ (10) Scale the indexes: (11)  $Inde x_1 = Inde x_1/L_{std}$ (12)  $Inde x_2 = -Inde x_2/CORT_{std}$ (13)  $Inde x_3 = Inde x_3/F_{std}(n)$ (14)  $Inde x_4 = Inde x_4/F_{std}(g)$ (15) Combine the indexes: (16)  $Score = Inde x_1 + Inde x_2 + Inde x_3 + Inde x_4$ (17) If score < = 0: (18) Input signals are normal. (19) Else:

(20) Input signals are abnormal.

$$W - \text{recall} = \frac{\sum_{i=1}^{s} (c_i \times \text{Recall}_i)}{\sum_{i=1}^{n} c_i},$$
  

$$W - \text{precision} = \frac{\sum_{i=1}^{s} (c_i \times \text{Precision}_i)}{\sum_{i=1}^{s} c_i},$$
(7)  

$$W - F = \frac{\sum_{i=1}^{s} (c_i \times F - \text{measure}_i)}{\sum_{i=1}^{s} c_i},$$

where  $c_i$  is the number of instances in class *i* and *s* is the number of total classes. Recall<sub>*i*</sub>, Precision<sub>*i*</sub>, and *F*-measure<sub>*i*</sub> are the recall, precision, and *F*-measure of the *i*th class, respectively.

4.2. Generation Capability of the Generator. To evaluate the performance of the generator, we compare the generation ability of the generator when the inputs are normal signals and abnormal signals. As we utilize only normal signals to train the MI-GAN, it is effective to obtain inverted noises z and produce similar signals to the normal signals. Nevertheless, the pretrained generator never learns the features of abnormal signals and thus cannot produce fake abnormal signals even when the inputs are abnormal, the generated signals will be dissimilar with input signals. Therefore, the generated signals will fall into two categories: one is similar signals whose inputs are abnormal signals.

Figure 5 presents the generation ability of our pretrained generator. In Figure 5, we input two types of signals (i.e., normal and abnormal ones) and utilize Algorithm 1 to

search specific noises *z*. Since the reconstruction capability for normal and abnormal signals is different, the generated signals will fall into two categories, i.e., similar signals and dissimilar signals. Using this characteristic of the generator, one can classify different types of input signals to achieve the tool wear conditions.

4.3. Discrimination Ability of the Discriminator. In this paper, instead of computing real or fake probability, the discriminator outputs a fake-level score, which can interpret how fake the signals are. To demonstrate the discrimination ability of the discriminator, we put the mean and standard deviation of the fake-level scores in Table 3.

According to Table 3, we can see that the mean fake-level scores of actual signals (-129.47 and -88.08) are lower than those of generated signals (-78.30 and -49.11), which indicates actual signals are more real than generated signals. In addition, normal signals have a lower mean fake-level score (-129.47) than that of abnormal signals (-88.08) since the generator can capture only the features of the normal signals. As for generated signals, it is straightforward to expect that the signals dissimilar to abnormal input signals have a higher mean fake-level score (-49.11) than that of the signals similar to normal input signals (-78.30). The standard deviation of abnormal signals is obviously larger than the remaining ones since the abnormal signals have more diverse modes than other ones.

4.4. Distribution of Four Indexes. To prove the effectiveness of four indexes, their respective distributions are plotted in Figure 6. We consider three different data types including normal signals in the training dataset, normal signals in the



FIGURE 5: Invert input signals to generate new signals. (a) If the input signal is normal, the generated signal is similar to the input signal. (b) If the input signal is abnormal, the generated signal is dissimilar to the input signal.

TABLE 3: The mean and standard deviation of the scores.

Types	Mean	Standard deviation
Scores of normal signals	-129.47	39.61
Scores of abnormal signals	-88.08	54.83
Scores of similar signals	-78.30	24.62
Scores of dissimilar signals	-49.11	29.41

testing dataset, and abnormal signals in the testing dataset (abbreviated as TraNS, TeNS, and TeAS) and present the corresponding distribution of each index. According to Figure 6, it can be seen that four indexes of TraNS and TeNS are extremely close, whereas four indexes of TraNS are different from those of TeAS. Therefore, we can exploit the mean and standard deviation of the indexes of TraNS to establish effective thresholds in the decision-making function so as to guarantee a satisfactory detection performance. 4.5. Effectiveness of Different Indexes. In this section, we explore the influence of different indexes on the performance of tool wear detection. For each individual index, we can calculate the corresponding value according to lines 11–14 of Algorithm 2 for anomaly detection. Furthermore, the performance of our MI-GAN is also presented as a comparison. In light of the results in Table 4, it can be observed that our MI-GAN has the best performance. The reason is that the proposed MI-GAN integrates multiple



FIGURE 6: Distribution of four indexes: (a) the  $L_2$ -norm index between generated signals and actual signals; (b) the CORT index between generated signals and actual signals; (c) the fake-level score of actual signals; (d) the fake-level score of generated signals.

TABLE 4: The comparison results of different index settings.

Index	W-recall	W-precision	W- $F$
L <sub>2</sub> -norm	0.81	0.83	0.82
CORT	0.83	0.85	0.84
Fake-level score of actual signals	0.87	0.91	0.88
Fake-level score of generated signals	0.83	0.88	0.85
MI-GAN	0.97	0.97	0.97

criteria and thus processes a more comprehensive detection capability.

4.6. Comparison with Other Machine Learning Methods. To further demonstrate the efficiency of the proposed MI-GAN, we compare it with other state-of-the-art machine learning methods: KNN [31], SVM [32], RF [33], CNN [14], Inception Net [34], DenseNet [35], ResNet [36], SE-Net [37], and Nonlocal-Net [38]. For a fair comparison, we reimplement those methods and adjust the hyperparameters on our training dataset to gain the best performance. The numerical results of different methods are illustrated in Table 5, from which we can conclude that the performance of CNN-based methods is better than that of the traditional machine learning ones (i.e., LNN, SVM, and RF). Besides, the proposed MI-GAN has a better performance with a significant improvement on accuracy, recall, precision, and F-measure compared with the CNN method. This is due to the fact that our MI-GAN could extract more critical features, and the excellent generation and discrimination

TABLE 5: The comparisons of different machine learning methods.

Method	W-recall	W-precision	W- $F$
KNN [31]	0.71	0.76	0.72
SVM [32]	0.67	0.75	0.69
RF [33]	0.69	0.76	0.71
CNN [14]	0.81	0.79	0.79
Inception Net [34]	0.85	0.89	0.83
DenseNet [35]	0.91	0.93	0.94
ResNet [36]	0.92	0.95	0.95
SE-Net [37]	0.93	0.96	0.95
Nonlocal-Net [38]	0.96	0.96	0.93
MI-GAN	0.97	0.97	0.97

capability of the MI-GAN could effectively handle the data imbalance challenge.

# 5. Conclusion

In this paper, we have developed a novel MI-GAN to detect the tool wear conditions for imbalanced industrial data where normal signals are much larger than abnormal ones. TCN residual blocks have been utilized as the backbone of the generator and discriminator networks to leverage the ability of feature extraction. In addition, a decision-making function with multi-index has been exploited to aggregate more representative criteria such that the detection performance can be further improved. For the sake of verifying the effectiveness and feasibility of our MI-GAN, different experiments have been conducted on the real-time current signals sampling from CNC machines. The experimental results have shown that our method indeed has an excellent performance on tool wear detection.

# **Data Availability**

The data used to support the findings of this study are available from the corresponding author upon request.

# **Conflicts of Interest**

The authors declare that they have no conflicts of interest.

#### Acknowledgments

This work was supported in part by the Key Project of Science and Technology Innovation 2030 supported by the Ministry of Science and Technology of China under Grant 2018AAA0101303, the National Natural Science Foundation of China under Grants 61803282 and 61973288, the National Key Research and Development Program of China under Grant 2018YFB2100801, the Fundamental Research Funds for the Central Universities of China under Grant 22120180561, the "Chenguang Program" Supported by the Shanghai Education Development Foundation and Shanghai Municipal Education Commission of China under Grant 18CG18, and the Shanghai Science and Technology Innovation Action Plan under Grants 20692193400 and 19511101300.

# References

- P. Bhattacharyya, D. Sengupta, S. Mukhopadhyay, and A. B. Chattopadhyay, "Online tool wear prediction in milling operation using feed motor current signal," *International Journal of Production Research*, vol. 46, no. 4, pp. 1187–1201, 2006.
- [2] S. Dutta, S. K. Pal, S. Mukhopadhyay, and R. Sen, "Application of digital image processing in tool condition monitoring: a review," *CIRP Journal of Manufacturing Science and Technology*, vol. 6, no. 3, pp. 212–232, 2013.
- [3] P. Waydande, N. Ambhore, and S. Chinchanikar, "A review on tool wear monitoring system," *Journal of Mechanical Engineering and Automation*, vol. 6, no. 5A, pp. 49–53, 2016.
- [4] Q. Hou, J. Sun, and P. Huang, "A novel algorithm for tool wear online inspection based on machine vision," *The International Journal of Advanced Manufacturing Technology*, vol. 101, no. 9–12, pp. 2415–2423, 2019.
- [5] D. Kerr, J. Pengilley, and R. Garwood, "Assessment and visualisation of machine tool wear using computer vision," *The International Journal of Advanced Manufacturing Technology*, vol. 28, no. 7-8, pp. 781–791, 2006.
- [6] P. Ong, W. K. Lee, and R. Lau, "Tool condition monitoring in CNC end milling using wavelet neural network based on machine vision," *The International Journal of Advanced Manufacturing Technology*, vol. 104, no. 1–4, pp. 1369–1379, 2019.
- [7] M. Hassan, A. Damir, H. Attia, and V. Thomson, "Benchmarking of pattern recognition techniques for online tool wear detection," *Procedia CIRP*, vol. 72, pp. 1451–1456, 2018.
- [8] D. Wu, C. Jennings, J. Terpenny, R. X. Gao, and S. Kumara, "A comparative study on machine learning algorithms for smart manufacturing: tool wear prediction using random forests,"

Journal of Manufacturing Science and Engineering, vol. 139, no. 7, p. 071018, 2017.

- [9] Y. He, M. Huang, and W. Sun, "Tool wear status recognition based on Mahalanobis distance," *The Journal of Engineering*, vol. 2019, no. 23, pp. 8802–8805, 2019.
- [10] B. Neef, J. Bartels, and S. Thiede, "Tool wear and surface quality monitoring using high frequency CNC machine tool current signature," in *Proceedings of the IEEE 16th International Conference on Industrial Informatics (INDIN)*, pp. 1045–1050, IEEE, July 2018, Porto, Portugal.
- [11] G. Simon and R. Deivanathan, "Early detection of drilling tool wear by vibration data acquisition and classification," *Manufacturing Letters*, vol. 21, pp. 60–65, 2019.
- [12] D. Wu, C. Jennings, J. Terpenny, and S. Kumara, "Cloudbased machine learning for predictive analytics: tool wear prediction in milling," in *Proceedings of the 2016 IEEE International Conference on Big Data (Big Data)*, pp. 2062–2069, IEEE, Washington, DC, USA, December 2016.
- [13] C. Zhang and H. Zhang, "Modelling and prediction of tool wear using LS-SVM in milling operation," *International Journal of Computer Integrated Manufacturing*, vol. 29, no. 1, pp. 76–91, 2016.
- [14] Y. Wang, W. Dai, and J. Xiao, "Detection for cutting tool wear based on convolution neural network," in *Proceedings of the* 2018 12th International Conference on Reliability, Maintainability, and Safety (ICRMS), pp. 297–300, IEEE, Shanghai, China, October 2018.
- [15] X.-C. Cao, B.-Q. Chen, B. Yao, and W.-P. He, "Combining translation-invariant wavelet frames and convolutional neural network for intelligent tool wear state identification," *Computers in Industry*, vol. 106, pp. 71–84, 2019.
- [16] A. Kothuru, S. P. Nooka, and R. Liu, "Application of deep visualization in CNN-based tool condition monitoring for end milling," *Procedia Manufacturing*, vol. 34, pp. 995–1004, 2019.
- [17] H. Zheng and J. Lin, "A deep learning approach for high speed machining tool wear monitoring," in *Proceedings of the 2019* 3rd International Conference on Robotics and Automation Sciences (ICRAS), pp. 63–68, IEEE, Wuhan, China, June 2019.
- [18] G. Martínez-Arellano, G. Terrazas, and S. Ratchev, "Tool wear classification using time series imaging and deep learning," *The International Journal of Advanced Manufacturing Technology*, vol. 104, no. 9–12, pp. 3647–3662, 2019.
- [19] L. E. E. Ochoa, I. B. R. Quinde, J. P. C. Sumba, A. V. Guevara Jr., and R. Morales-Menendez, "New approach based on autoencoders to monitor the tool wear condition in HSM," *IFAC-PapersOnLine*, vol. 52, no. 11, pp. 206–211, 2019.
- [20] X. Cao, B. Chen, B. Yao, and S. Zhuang, "An intelligent milling tool wear monitoring methodology based on convolutional neural network with derived wavelet frames coefficient," *Applied Sciences*, vol. 9, no. 18, p. 3912, 2019.
- [21] W. Jiang, Y. Hong, B. Zhou, X. He, and C. Cheng, "A GANbased anomaly detection approach for imbalanced industrial time series," *IEEE Access*, vol. 7, pp. 143608–143619, 2019.
- [22] D. Li, D. Chen, J. Goh, and S. K. Ng, "Anomaly detection with generative adversarial networks for multivariate time series," 2018, https://arxiv.org/abs/1809.04758?.
- [23] T. Schlegl, P. Seeböck, S. M. Waldstein, U. Schmidt-Erfurth, and G. Langs, "Unsupervised anomaly detection with generative adversarial networks to guide marker discovery," *Lecture Notes in Computer Science*, Springer, Cham, Switzerland, pp. 146–157, 2017.
- [24] A. D. Chouakria and P. N. Nagabhushan, "Adaptive dissimilarity index for measuring time series proximity,"

Advances in Data Analysis and Classification, vol. 1, no. 1, pp. 5–21, 2007.

- [25] S. Bai, J. Z. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," 2018, https://arxiv.org/pdf/1803.01271.pdf.
- [26] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein gan," 2017, https://arxiv.org/abs/1701.07875.
- [27] J. Wu, Z. Huang, J. Thoma, D. Acharya, and L. Van Gool, "Wasserstein divergence for GANS," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 653–668, Munich, Germany, September 2018.
- [28] D. P. Kingma and J. Ba, "Adam: a method for stochastic optimization," 2014, https://arxiv.org/abs/1412.6980.
- [29] A. Creswell and A. A. Bharath, "Inverting the generator of a generative adversarial network," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 7, pp. 1967–1974, 2018.
- [30] S. Ruder, "An overview of gradient descent optimization algorithms," 2016, https://arxiv.org/abs/1609.04747.
- [31] Z. Deng, X. Zhu, D. Cheng, M. Zong, and S. Zhang, "Efficient k NN classification algorithm for big data," *Neurocomputing*, vol. 195, pp. 143–148, 2016.
- [32] J. A. K. Suykens and J. Vandewalle, "Least squares support vector machine classifiers," *Neural Processing Letters*, vol. 9, no. 3, pp. 293–300, 1999.
- [33] J. Cao and G. Fan, "Signal classification using random forest with kernels," in *Proceedings of the 2010 Sixth Advanced International Conference on Telecommunications*, pp. 191– 195, IEEE, May 2010, Barcelona, Spain.
- [34] C. Szegedy, W. Liu, Y. Jia et al., "Going deeper with convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–9, Boston, MA, USA, June 2015.
- [35] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings* of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4700–4708, Honolulu, HI, USA, July 2017.
- [36] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference* on Computer Vision and Pattern Recognition, pp. 770–778, Las Vegas, NV, USA, June 2016.
- [37] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7132–7141, Salt Lake City, UT, USA, June 2018.
- [38] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7794–7803, Salt Lake City, UT, USA, June 2018.