

CNN-GRU Network-based Force Prediction Approach for Variable Working Condition Milling Clamping Points of Deformable Parts

Enming Li (✉ eastwoodlee@foxmail.com)

School of Mechanical Engineering, Northwestern Polytechnical University, Xi'an, 710072, China)

<https://orcid.org/0000-0001-6477-8426>

Jingtao Zhou

School of Mechanical Engineering, Northwestern Polytechnical University, Xi'an, 710072, China)

Changsen Yang

School of Mechanical Engineering, Northwestern Polytechnical University, Xi'an, 710072, China)

Mingwei Wang

School of Mechanical Engineering, Northwestern Polytechnical University, Xi'an, 710072, China)

Zeyu Li

School of Mechanical Engineering, Northwestern Polytechnical University, Xi'an, 710072, China)

Huibo Zhang

School of Mechanical Engineering, Northwestern Polytechnical University, Xi'an, 710072, China)

Tengyuan Jiang

School of Mechanical Engineering, Northwestern Polytechnical University, Xi'an, 710072, China)

Research Article

Keywords: Clamping point force prediction, Variable working conditions, Convolutional neural network, Gated recurrent unit, Temporal correlation, Deformable parts

Posted Date: June 25th, 2021

DOI: <https://doi.org/10.21203/rs.3.rs-636352/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Version of Record: A version of this preprint was published at The International Journal of Advanced Manufacturing Technology on January 28th, 2022. See the published version at <https://doi.org/10.1007/s00170-021-08520-2>.

CNN-GRU Network-based Force Prediction Approach for Variable Working Condition Milling Clamping Points of Deformable Parts

Enming Li^{1*}, Jingtao Zhou^{1*}, Changsen Yang¹, Mingwei Wang¹, Zeyu Li¹, Huibin Zhang¹, Tengyuan Jiang¹
(1 School of Mechanical Engineering, Northwestern Polytechnical University, Xi'an, 710072, China)

Abstract: Improper clamping is one of the major causes of part deformation. Improving the fixture arrangement through force analysis of clamping points is an effective means to suppress or improve machining deformation. However, the existing research focuses on the monitoring and off-line optimization of the clamping point force, which has a certain lag on the machining deformation control, and it is difficult to predict the clamping point force due to the time-varying coupling effect of multiple factors such as process parameters, cutting force and clamping point force in the machining process. Inspired by the excellent performance of convolutional neural networks and gated recurrent networks in feature extraction and learning of temporal association laws, this paper proposes a CNN-GRU-based method for predicting the force state of clamping points under variable working conditions. Firstly, a force prediction model of clamping point during milling process with variable working conditions is established. Secondly, a convolutional neural network is designed to extract the features of dynamic coupled machining conditions. Then, a network of gated recurrent units is constructed to learn the temporal correlation law between the machining conditions and the forces on the clamping points to achieve force prediction of the clamping points during machining. Finally, it was verified by the milling process of the piston skirt. The results show that CNN-GRU can effectively predict the clamping force. In addition, CNN-GRU has higher computational efficiency and accuracy compared with CNN-LSTM, CNN-RNN and CNN-BP.

Keywords: Clamping point force prediction, Variable working conditions, Convolutional neural network, Gated recurrent unit, Temporal correlation, Deformable parts

1 Introduction

Part processing deformation is a common problem in CNC machining, such as the machining deformation of large-size integral parts in aerospace industry, diesel engine piston skirt, automobile impeller, etc., which seriously affects the machining accuracy and surface quality of the workpiece [1, 2]. Deformation caused by external loads (clamping, cutting forces) that elastically deform the workpiece during processing, being recovered afterwards [3]. This means that the fixture becomes a key component to avoid the geometrical error associated to deformation caused by external loads during machining processes. The fixture is a precision subsystem to provide an accurate positioning of the workpiece in the work space and to rigidly hold and support the component to withstand the machining forces, affecting the static and dynamic behaviour of the workpiece [4]. The deformation of the workpiece is not only related to the residual stress and cutting force of the workpiece, but also related to the processing conditions such as cutting order, fixture arrangement, clamping force, etc. [5], in which the deformation caused by the clamping of the fixture is usually related to the existing deformation caused by the previous manufacturing process and clamping force in the original workpiece, especially in the processing of thin-walled parts, the larger clamping force will lead to excessive elastic deformation of the workpiece, resulting in larger dimensional errors, while the smaller clamping force cannot adequately restrain the workpiece during the processing. Therefore, accurate prediction of clamping point force can help to correct fixture arrangement and clamping force, and then effectively restrain or improve machining deformation. From this perspective, it is a good entry point to study the prediction of

* Corresponding author. Tel.: +86 29 88493232; Fax: +86 29 88493232.
E-mail address: eastwoodlee@foxmail.com (Enming. Li).

* Corresponding author. Tel.: +86 29 88493232; Fax: +86 29 88493232.
E-mail address: zhoujt@nwpu.edu.cn (Jingtao. Zhou).

clamping point force during the machining of thin-walled parts. However, the variation of clamping point force is influenced by a combination of several factors such as cutting parameters, cutting force, and vibration during machining, and these factors show complex coupling changes in time sequence as the working conditions change, making it a challenge to accurately predict the clamping point force under variable working conditions.

At present, the analysis or prediction methods for the machining process clamping point force mainly include different modeling methods such as analytical method and finite element method to numerically simulate the clamping process and its deformation [6, 7], and these works are the basis for the study of fixture layout optimization. However, these models rely on input parameters that are difficult to obtain or estimate, such as the clamping point forces during machining, and a large number of assumptions are required to ensure that the model can be solved or approximate solutions are obtained. Due to the uncertainties in the machining process, the simulation process differs from the actual machining process, and the clamping point forces are constantly changing under the complex coupling of cutting parameters, cutting forces, vibrations, and other operating conditions, off-line clamping Optimization or fixed clamping simulation model can not adapt to such changes, and it is not possible to optimize the fixture arrangement and clamping force size, as well as further processing deformation control according to the changes in clamping point forces.

Nowadays, sensors and monitoring technologies have evolved significantly allowing the on-line control of the actual state of the manufacturing systems [8]. Some studies use sensors, including displacement sensors or strain gauges, in fixtures and clamping elements to monitor clamping and reaction forces and design dynamically adjustable fixtures based on this, such as dynamic fixtures by monitoring clamping forces [4], or floating clamping methods based on real-time displacement monitoring data [9, 10]. The core principle of the above research is to adjust the fixture state by monitoring the change of force or displacement at the clamping point of the machining process, so as to achieve the purpose of releasing the machining stress of the workpiece and controlling the machining deformation, which means that the correction of the clamping state by analyzing the change of force at the clamping point becomes one of the effective ways to suppress or control the machining deformation.

However, the current use of fixtures or workpiece-fixture systems to analyze machining deformation is more a matter of converting machining deformation prediction into online monitoring and detection of workpiece deformation during machining, and less work has been done to directly predict the force state at the clamping point. Moreover, the uncertainty of the actual machining process leads to the fact that the work condition information (e.g. cutting force) is often missing to some extent or contains a lot of noisy data and redundant information, which is difficult to be used directly as a basis for adjusting the clamping state. Moreover, there is a time-varying coupled influence relationship between the working condition factors and the clamping force state, and the accuracy of the clamping point force prediction method relies on the extraction of the dynamic time-series characteristics related to the clamping force from the complex changing working conditions, and the existing research cannot fully explore the time-series correlation between the time-varying working conditions and the clamping point force.

The rapid development in the field of computer science such as machine learning, deep learning, provides new data-driven ideas for discovering patterns in machining processes from these data [11], especially for complex, dynamic and even chaotic manufacturing processes reflecting great advantages, such as applications in tool condition monitoring [12], machining accuracy prediction [13], and machining deformation prediction problems [14]. Inspired by this, this paper proposes a CNN-based feature extraction of time-varying working conditions of machining process, with GRU-based learning of dynamic time-series relationship between variable working conditions and clamping point force for clamping point force prediction. The main contributions of this study are summarized as follows:

- 1) We propose the use of hierarchical mean interpolation, Donoho threshold and soft-hard threshold compromise method and Z-score normalization method to pre-process the time-varying working condition data according to the characteristics of continuous change of time-varying working condition data, and compare and analyze the effect of different data processing methods.
- 2) We designed a time-varying working condition feature adaptive extraction model with a fully convolutional CNN network to realize the feature extraction of high-dimensional sensitive data responding to the change of force state at the clamping point.
- 3) A GRU network-based force state prediction model for clamping points is proposed, and several different prediction models are compared. Experimental results demonstrate the effectiveness of the CNN-GRU model that considers dynamic time-series correlation.

The remainder of this paper is organized as follows: Section 2 reviews the existing approaches used for force prediction at clamping points. In Section 3, the factors affecting the clamping point force are analyzed and some definitions and framework overview of CNN-GRU based clamping point force prediction methods are presented. Section 4 gives details of the time-varying operating condition data preprocessing, feature extraction, and clamping point force prediction model learning methods. In Section 5, piston skirt milling machining is used as an experimental object and compared with other models for analysis to verify the performance of the proposed model. Finally, Section 6 summarizes the whole paper and gives an outlook on future research directions.

2 Related work

In this section, we briefly review three research fields that are relevant to this paper. Specifically, the three aspects include force state prediction, time-varying data feature extraction, and recurrent neural network application.

2.1 Prediction of force state in machining process

The methods for force prediction during processing can be broadly classified into three categories: mechanical analysis modeling method, finite element simulation method, and machine learning method. Among them, the mechanical analysis modeling mainly refers to the theoretical analysis of the forces and related parameters in the processing process, to establish the corresponding theoretical numerical model. Martin et al. [15] proposed a systematic mathematical approach to optimize these workpiece clamping parameters using an analytical model of ring deformation and a model to determine the minimum clamping force. Using the principle of minimum modulus, Zhang et al. [16] developed a kinematic model of the workpiece-fixturing system, calculated the contact forces (including frictional forces) between the fixture components and the workpiece, and further proposed a model to optimize the clamping force to maintain the stability of the workpiece-fixturing system. Finite element simulation method refers to offline machining simulation prediction using commercial finite element simulation software. A machining surface error prediction method that considers the dynamic effects of clamping and workpiece systems during milling was proposed by Dong et al. [17]. The static deformation caused by the clamping force and the dynamic deformation caused by the milling force are calculated using static analysis and explicit dynamic analysis methods in the FEA environment. Liu et al. [18] proposed a milling error prediction method based on the finite element method, which considered the deflection of the workpiece/tool system and the springback deformation of the workpiece, in addition to the dynamics model of the tool. Machine learning-based force state prediction for machining processes mainly uses artificial neural networks to learn the correlation between the force state and the relevant machining parameters. Zhao et al. [14] proposed a deep learning model-based online prediction method for CNC machined part deformation, using deformation monitoring data and machining information related to intermediate part geometry information to construct and train a deep learning model consisting of traditional neural networks

and recurrent neural networks to achieve online prediction of part deformation. Teramoto et al. [19] evaluated the deformation of workpieces under different clamping sequences, using locally measured strains and clamping simulations, and used the response surface method to estimate clamping forces and workpiece deformation. Hao et al. [20] designed a responsive fixture that can make responsive adjustments to part deformation based on online monitoring data, and optimize the machining sequence through clustering and dynamic planning to reduce machining deformation due to unpredictable factors.

Although the above three methods have achieved certain results, but theoretical mechanical analysis modeling and finite element simulation to describe the complex manufacturing process is limited, the analysis process inevitably contains simplifications, and sometimes contains inappropriate assumptions, and therefore difficult to adapt to the actual machining process of time-varying work factors, the dynamic coupling between work factors analysis process also has a gap with the actual machining. Machine learning methods provide ideas for mining the correlation between the force state and the working condition of the machining process, but the traditional neural network structure level is relatively too shallow, and the ability to learn the complex nonlinear relationship between the time-varying working condition and the force at the clamping point is relatively weak.

2.2 Time-varying processing data feature extraction

The existing time-varying data feature extraction methods can be broadly classified as traditional feature extraction methods and deep learning-based feature extraction methods. Traditional feature extraction methods can be divided into three categories based on the implementation principle: time domain, frequency domain, and time-frequency domain, where time-domain feature extraction refers to obtaining various time-domain parameters or indicators in signal data through statistical methods, such as Wu et al. [21] performed tool remaining life prediction by calculating and filtering parameters such as standard deviation, mean value, root mean square value, root square amplitude, and maximum value of noise signals and current signals. And frequency domain feature extraction usually refers to the use of Fourier transform for frequency component analysis of signal data, such as the frequency amplitudes in acoustic emission signals and vibration signals obtained by Wang et al. [22] using fast Fourier transform as part of the input to the tool wear prediction model. But the actual acquisition of processing dynamic signal is often non-smooth, only one-sided analysis of its time domain or frequency domain characteristics is not enough, but also to further clarify the signal spectrum changes with time, so the need to use the joint time-frequency function of the signal data for time-frequency domain analysis. Zhou et al. [23] used HHT transform for feature extraction of torque signals and implemented tool wear life prediction. Compared with the traditional feature extraction methods, the deep learning feature extraction method uses a multi-hidden layer machine learning model to learn autonomously to obtain useful data features, such as Du et al. [24] proposed a 3D convolutional neural network-based spatio-temporal feature learning method.

Compared with traditional time domain, frequency domain, and time-frequency domain data feature extraction methods, deep learning can deeply explore the high-dimensional feature information hidden in the data, significantly eliminate the influence of subjective experience on the feature extraction process, and reduce the loss of information features caused by human factors, the advantage of this unsupervised adaptive extraction of features makes it more effective in dealing with complex processing conditions feature extraction. In this paper, we consider multiple time-varying operating conditions data as images composed of single-channel pixel matrices, and then use CNN networks to adaptively extract time-varying operating conditions features at each moment (or time period).

2.3 Recurrent neural network

Recurrent neural network (RNN) was first proposed to process sequential data in the 1980s [25]. However, due to the problems of gradient disappearance and gradient explosion, the original RNN is very difficult to be trained, and its application is very limited. To tackle these problems, Schmiduber et al. [26] proposed the long short-term memory (LSTM) which is a variant of RNN, and then on this basis Jiang et al. [27] proposed the gated recurrent unit (GRU) network. Although both of them are RNN networks, the former is a recurrent neural network with a special structure, while the latter belongs to a variant of the former structure. They both solve the problems of gradient explosion, gradient disappearance, and insufficient long-term memory capacity of traditional recurrent neural networks while introducing the temporal feedback mechanism, and thus effectively utilize the previous temporal information and improve the model reliability. LSTM has been successfully used for anomaly detection in mechanical equipment [28], the authors' team [12, 23] also proposed the use of LSTM to predict the tool wear state under different working conditions. Compared with the standard LSTM network, the GRU network structure is more streamlined and computationally simpler. Zhao et al. [29] implemented the monitoring of three machine health states (tool wear prediction, gearbox fault diagnosis, and early bearing fault detection) by constructing a local feature-based gated recursive unit (LFGRU) network. It can be seen that recurrent neural networks have significant advantages in dealing with complex time-series change prediction problems during processing. In this paper, a GRU network-based force prediction method for clamping points is proposed, taking into account the complex variation of machining conditions and clamping point force states during machining and the correlation between the two in time sequence

3 Basic concepts and overview of the proposed approach

In this section, we first introduce the basic concepts related to this paper, and then the framework and process of force prediction at the clamping point is given.

3.1 Basic concepts

Definition 1 Clamping point force (F). The clamping point force represents the force that the fixture is subjected to under the action of external and internal forces (workpiece processing deformation, cutting force, etc.) acting on the workpiece. Clamping point force can be denoted as $F = (F_x, F_y, F_z)$, and F_x, F_y, F_z denote the component forces in the X, Y and Z directions respectively.

Definition 2 Machining conditions (C). The machining conditions are a collection of factors that affect the force on the clamping point during machining. The machining conditions can be denoted as $C = \{C_{pr}, C_{cl}, C_{mt}, C_{tool}, C_{part}, C_{mon}\}$, and $C_{pr}, C_{cl}, C_{mt}, C_{tool}, C_{part}, C_{mon}$ indicate the machining sub-conditions, clamping sub-conditions, machine sub-conditions, tool sub-conditions, workpiece sub-conditions and monitoring signal sub-conditions respectively.

Definition 3 Sampling interval (S_v). The sampling interval is the basis for intercepting the processing conditions in the time sequence, which can ensure the correspondence and consistency of various working conditions factors in the time sequence within any time segment.

Definition 4 Time window (T_w). Time windows are used to intercept data samples in Definition 3 with different time lengths to form the corresponding training and prediction samples to support subsequent data pre-processing and network training.

Definition 5 Problem (Clamping point force prediction). The force prediction at the clamping point is to output the force state at the clamping point after t hours of continued machining, given the machining condition data for a period of time T , after data pre-processing and feature fusion operations, and input to the prediction model. That is $Input=C$, after data pre-processing operation and feature extraction operation, the processing results are then fed into the prediction function Y , and the output result is as follows $Output = \{(F_x)_{T+t}, (F_y)_{T+t}, (F_z)_{T+t}\}$.

3.2 Overview of the proposed approach

The procedures of the proposed method are illustrated in Figure. 1. It involves four parts: data pre-processing, feature extraction, network training and evaluation, clamping point force prediction. Here, we give a brief description of each part, respectively:

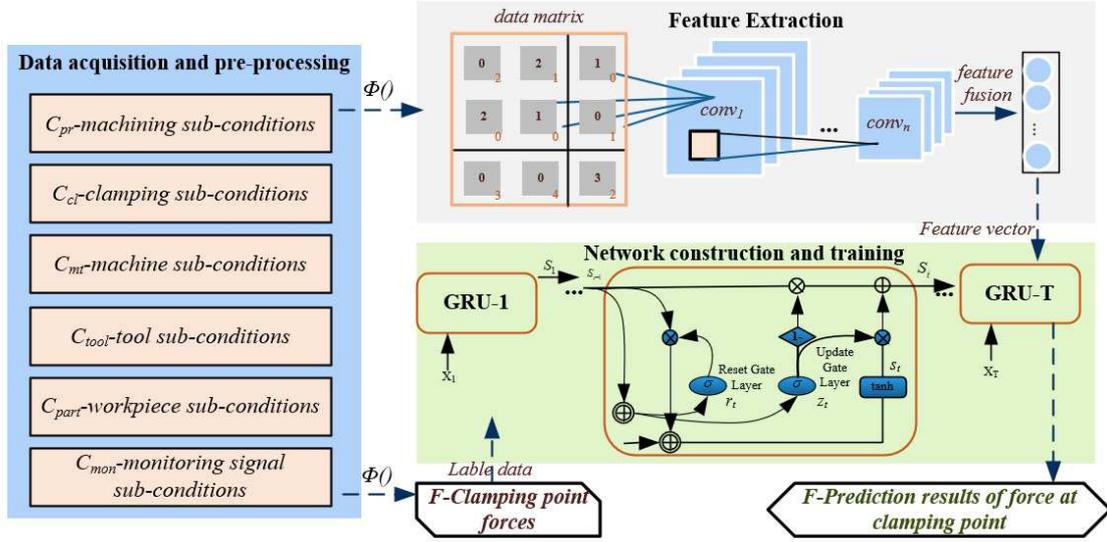


Figure. 1. The framework of the clamping point force prediction based on CNN-GRU

(1) Data pre-processing

The pre-processing of different working condition data during processing is an essential step, because the raw data are usually subject to local extremes, missing data, data noise and inconsistency in the data scale due to the actual processing environment and data acquisition conditions, which seriously affect the subsequent data analysis and model construction results. In this paper, the data pre-processing operations Φ include missing value processing, noise reduction and normalization.

(2) Feature extraction

Considering that the processing process involves a variety of time-varying factors, the data is often characterized by large quantities, dynamic changes and strong coupling, so in order to extract the feature information efficiently and reliably, this paper treats the time-varying data as a single-channel image composed of two-dimensional pixel matrix, and then builds a CNN network without pooling operations for unsupervised adaptive feature extraction to obtain training samples.

(3) Network training and evaluation

The GRU model from the bottom up mainly consists of three parts: prediction model input, prediction model forward calculation, and prediction model backward tuning. The processing conditions in each time window of the current time period are first fused using the Concat algorithm, and then used as input to the prediction model. The mean square error (MSE) is used as the loss function in the reverse tuning process, and the gradient descent process is optimized using Adam's algorithm. After model training, we quantitatively evaluate the fit of the model predictions to the true values through a comparative analysis of several prediction models.

(4) Clamping point force prediction

In this step, the new machining conditions data are inputted into the prediction function Y , and then output a vector of three-way clamping force values at the clamping point.

The following sections describe data pre-processing, feature extraction, networks construction and training process in more detail.

4 Methodology

In this section, we first introduce the preprocessing of the time-varying working condition data of the machining process, and then state the extraction of force-sensitive features at the clamping

point. Finally, we elaborate the process of our proposed force prediction model for the clamping point.

4.1 Pre-processing of machining condition data

In order to improve the quality of the extracted data features, we need to pre-process the raw data, i.e. remove the redundant data, supplement the missing data and unify the inconsistent data through certain data processing methods, and then lay the foundation for the construction of the subsequent time-varying working condition feature extraction model and clamping force state prediction model.

(1) Missing value handling

For the characteristics of time-varying machining conditions, we choose the layered mean interpolation method to supplement the missing values. The method of stratified mean interpolation solves the problem well by first finely stratifying the variables according to their attribute characteristics before interpolation to ensure that the data in each stratum have similar characteristics, and then treating the mean value of the data in the missing cells in each stratum as the interpolated value in the missing cells in that stratum.

The interpolation values under the hierarchical mean interpolation method are calculated as follows:

$$\bar{y}_k = \frac{1}{n} \sum_{i=1}^n a_i y_i \quad (1)$$

Where \bar{y}_k is the interpolation value of the k th layer; a_i proxy for the presence of missing data at the i -th cell in the k -th layer, $a_i = 1$ means no missing data, $a_i = 0$ means missing data; y_i is the data value at the i -th cell; n is the number of data cells in the k -th layer.

(2) Wavelet threshold denoising

Due to the influence of machining environment, machine tool system or sensor performance, there are usually many noisy data in the collected time-varying working condition data, and the redundant information contained in these noisy data will increase the data analysis calculation and affect the model accuracy, so it is necessary to adopt a reasonable and feasible method to noise reduction and filtering of the original data. The ideal noise reduction filtering method should remove data noise while ensuring minimum data distortion. Based on the principle of wavelet transform and the good application effect of wavelet threshold denoising method [30, 31], We choose to use wavelet threshold denoising (WTD) method for noise reduction and filtering of the original time-varying working condition data.

Firstly, select the suitable wavelet base and decomposition layers, set the threshold value and threshold function; then input the original data to be processed and carry out the corresponding wavelet decomposition, keep all the wavelet coefficients of the decomposed low frequency part, compare the wavelet coefficients of the high frequency part with the set threshold value, if the wavelet coefficients are greater than the threshold value, keep them or carry out the corresponding computational transformation, if the wavelet coefficients are less than the threshold value, set them to zero. Finally, the filtered wavelet coefficients are used to reconstruct the time-varying data by wavelet inverse transform to obtain high-quality data. In the whole processing process, the threshold value and the setting of the threshold function have a decisive influence on the effect of noise reduction.

For wavelet threshold denoising methods, there are two commonly used threshold calculation methods at this stage. One is the generic threshold given by Donoho et al. which is always constant across scales of the decomposition and is calculated by the following equation:

$$\lambda = \sigma \sqrt{2 \log(N)} \quad (2)$$

Where σ is the noise intensity; N is the signal length.

The other is the improved threshold of Zhang et al.[32]. The threshold decreases as the data

decomposition scale increases, which is consistent with the transfer variation characteristics of wavelet coefficients at different decomposition scales. It is calculated by the following equation:

$$\lambda(j) = \sigma \sqrt{2 \log(N) / \log(j+1)} \quad (3)$$

Where σ is the noise intensity; N is the signal length; j is the decomposition scale

And the corresponding threshold function construction methods can be broadly divided into three.

The first is the hard threshold method, which directly sets the wavelet coefficients to zero when the wavelet decomposition is less than the set threshold, and directly retains them when they are greater than the set threshold, with the following formula:

$$w_{j,k} = \begin{cases} w_{j,k} & |w_{j,k}| \geq \lambda \\ 0 & |w_{j,k}| < \lambda \end{cases} \quad (4)$$

Where $w_{j,k}$ is the wavelet coefficient; λ is the set threshold value.

The second is the soft threshold method, which directly sets the wavelet coefficients to zero when the wavelet decomposition is less than the set threshold, and shrinks them to zero by a fixed amount when it is greater than the set threshold, with the following formula:

$$w_{j,k} = \begin{cases} \text{sgn}(w_{j,k})(|w_{j,k}| - \lambda) & |w_{j,k}| \geq \lambda \\ 0 & |w_{j,k}| < \lambda \end{cases} \quad (5)$$

Where $w_{j,k}$ is the wavelet coefficient; λ is the set threshold value; sgn is a symbolic function.

The last one is the soft and hard threshold compromise method. Since the wavelet coefficients of the low frequency part obtained from each decomposition will always be affected by the noise wavelet coefficients, this method reduces this effect by increasing the adjustment coefficients, with the following formula:

$$w_{j,k} = \begin{cases} \text{sgn}(w_{j,k})(|w_{j,k}| - a\lambda) & |w_{j,k}| \geq \lambda \\ 0 & |w_{j,k}| < \lambda \end{cases} \quad (6)$$

Where $w_{j,k}$ is the wavelet coefficient; λ is the set threshold value; sgn is a symbolic function; a is the adjustment coefficient, and its value range is $[0,1]$.

To determine the most effective wavelet threshold noise reduction method, this paper takes the time-varying working condition data in the piston skirt milling data set, i.e., 200 consecutive sampling points in the machine tool vibration signal (Y direction) as an example, and determines the wavelet base (*db3* wavelet), the number of decomposition layers ($j=1$), the signal length ($N=2$), and the adjustment coefficient ($a=0.5$) on the basis of both qualitative and quantitative levels. The noise reduction filtering performance of different threshold values and combinations of threshold functions are compared and analyzed.

a) Qualitative analysis

Qualitative analysis is mainly from the perspective of noise reduction filtered signal image comparison analysis Donoho threshold+hard threshold method, Donoho threshold+soft threshold method, Donoho threshold+hard and soft threshold compromise method, Zhang threshold+hard threshold method, Zhang threshold+soft threshold method, Zhang threshold+hard and soft threshold compromise method under the six combinations of machine tool vibration signal noise reduction filtering effect, different combinations of noise reduction filtering results are shown in Figure. 2 below.

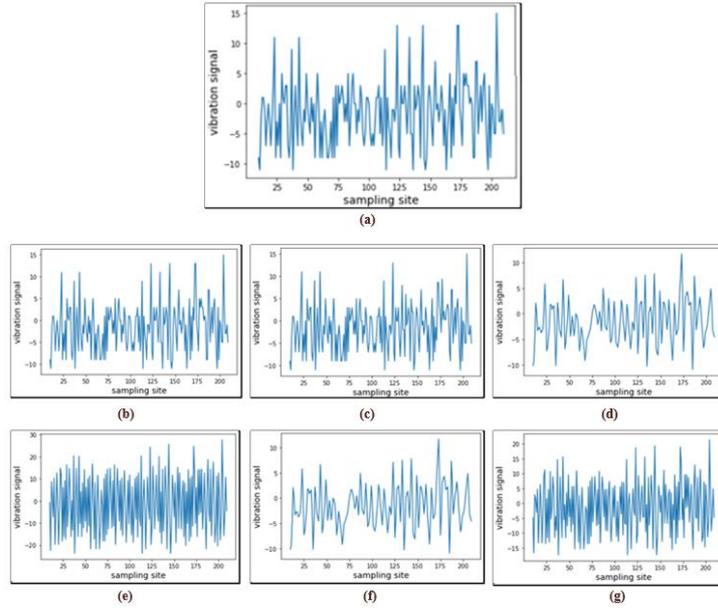


Figure. 2. (a) Raw vibration signal data. (b) Donoho threshold+hard threshold method. (c) Zhang threshold+hard threshold method. (d) Donoho threshold+soft threshold method. (e) Zhang threshold+soft threshold method. (f) Donoho threshold+soft and hard threshold compromise method. (g) Zhang threshold+ soft and hard threshold compromise method.

It can be seen that compared with the original vibration signal data curve, the vibration signal data curve in (b) has not changed significantly, and the noise reduction filtering effect is the worst; the vibration signal data curve in (c) has changed slightly, and its smoothness and volatility have improved slightly but the overall effect is not good; the vibration signal data curves in (e) and (g) have improved greatly in terms of smoothness, but the vibration signal volatility has increased after filtering, and its fluctuation interval has increased instead of decreased; the vibration signal data curves in (d) and (f) have improved greatly in terms of smoothness and vibration signal volatility, and the noise reduction filtering effect is relatively better. In (d) and (f), the vibration signal data curves have improved greatly in terms of smoothness and vibration signal volatility, and the data noise reduction filtering effect is relatively better, but it is difficult to judge the performance of both from the signal plots. Therefore, based on the qualitative analysis, it is necessary to quantitatively analyze the noise reduction filtering ability of wavelets with six different threshold values and combinations of threshold functions.

b) Quantitative comparison

We quantitatively measure the noise filtering performance of the above six methods by calculating the speckle noise index β of the noise reduction filtered data. The index is the ratio of the standard deviation of the filtered data to the mean value, and usually the smaller the β value is, the better the noise filtering performance is. Suppose the original machine vibration signal data is $X = \{x_1, x_2, \dots, x_n\}$, the noise reduction filtering process to get the data is $X' = \{x'_1, x'_2, \dots, x'_n\}$, at this time its speckle noise index is calculated by the following formula:

$$\beta = \frac{\sqrt{\frac{1}{n} \sum_{i=1}^n (x'_i - \frac{1}{n} \sum_{i=1}^n x'_i)^2}}{\frac{1}{n} \sum_{i=1}^n x'_i} \quad (7)$$

Using the formula to quantitatively analyze the noise reduction and filtering performance of machine tool vibration signals under the above six combinations, the specific results are shown in the following Table 1:

Table 1 β -values under different combinations

combinations	β value
Donoho threshold+hard threshold	-0.0445
Zhang threshold+hard threshold	-0.0893
Donoho threshold+soft threshold	-0.5297
Zhang threshold+soft threshold	-0.2419
Donoho threshold+soft and hard threshold compromise	-0.5331
Zhang threshold+ soft and hard threshold compromise	-0.2760

As can be seen from the table, when the threshold value is selected as Donoho threshold and the threshold function construction method is selected as soft and hard threshold compromise method, the β -value is the smallest, i.e., the machine tool vibration signal data is filtered relatively better. Based on the above qualitative and quantitative analysis results, we choose to use Donoho threshold and soft-hard threshold compromise method for noise reduction and filtering of time-varying working condition data.

(3) Data standardization

The data of machining conditions often exist in different levels or units, and it is necessary to standardize these data to eliminate the influence of levels between different index parameters and solve the problem of incomparable data indicators. The main normalization methods commonly used are Z-score, Min-max, nonlinear. Considering that the time-varying machining data (such as cutting force signal data, machine vibration data, etc.) processed in this paper have positive and negative variations, the Z-score normalization method is directly chosen to process the data.

Z-score normalization is mainly calculated by means of the mean and standard deviation in the data set, and the data normalized using this method conform to a standard normal distribution, i.e., a mean of 0 and a standard deviation of 1. For a data set $X=\{x_1, x_2, \dots, x_n\}$, for any $x \in X$, the transformation is calculated as follows:

$$x' = \frac{x - \mu}{\sigma} \quad (8)$$

Where x' is the normalized z-score; μ is the mean of the samples in the dataset; and σ is the standard deviation of the samples in the dataset.

4.2 Machining conditions feature extraction

In this section, the time-varying machining condition data in time window Tw is treated as a single-channel pixel matrix image, and then the CNN network is used to extract the condition features adaptively. However, considering that the CNN network can not perform pooling operation when processing small image blocks, this paper treats the time-varying working condition data matrix as small pixel images (i.e., small image blocks), so the feature extraction model does not do the corresponding pooling operation after the convolution operation.

This paper introduces the feature extraction process using the example of piston skirt milling process data containing three types of time-varying working conditions (X/Y/Z three-way machine vibration data, X/Y/Z three-way force data at the clamping point, and X/Y/Z three-way cutting force data). The network structure of the feature extraction model is shown in the Figure. 3 below:

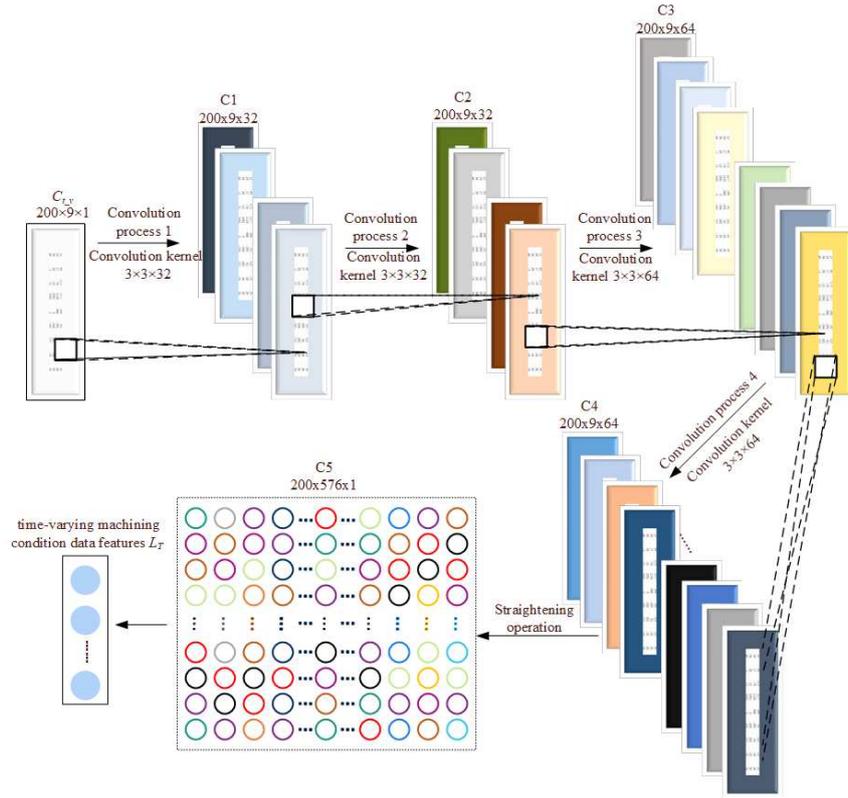


Figure 3. CNN network-based feature extraction for time-varying working condition data

The sample data of time-varying machining conditions ($C_{t,v}$) forms a 200×9 time-varying working condition data matrix after interval sampling S_v . The specific process is shown in the following Table 2:

Table 2 Feature extraction process

<i>Input:</i> After data pre-processing, the machining condition data sample $C_{t,v}$ in time T ;
<i>Output:</i> Data feature vector L_T
Step1: The original time-varying working condition data $C_{t,v}$ are subjected to interval sampling, prediction sample division, time window division, data pre-processing and other operations to obtain the data matrix $(C_{t,v})_T$ in time T ;
Step2: Convolution process 1 is performed on this data matrix using 32 kinds of $3 \times 3 \times 1$ convolution kernels to obtain 32 new 200×9 -dimensional data feature matrices. The number of trainable parameters in the convolution kernel, the number of neurons in the $C1$ layer, and the total number of connections are shown in the figure above;
Step3: The 32 new 200×9 -dimensional data feature matrices are obtained by convolution process 2 using 32 $3 \times 3 \times 32$ convolution kernels on the data matrix $(C_{t,v})_T$ obtained from the first abstraction extraction. This process is the second abstraction of the hidden features in the data matrix;
Step4: Convolution process 3 is performed on the data matrix obtained from the second abstraction with 64 $3 \times 3 \times 32$ convolution kernels to obtain 64 new 200×9 -dimensional data feature matrices. This process is the third abstraction to extract the hidden features from the data matrix $(C_{t,v})_T$.
Step5: As above, the convolution process4 is performed on the data matrix obtained from the third abstraction extraction using 64 kinds of $3 \times 3 \times 64$ convolution kernels to obtain 64 new 200×9 -dimensional data feature matrices. This process is the fourth abstraction extraction of the hidden features in the data matrix;
Step6: The 64 200×9 -dimensional data feature matrices extracted by the fourth abstraction are pulled into a 200×576 -dimensional data matrix in order;
Step7: The data matrix obtained from Step6 is used as input, and the force value data f_{T+t} in the three directions of X/Y/Z at the clamping point $T+t$ is used as the label, and the parameters in the model are updated and adjusted using the principle of reverse error transmission, and the final 200×576 dimensional data matrix obtained is the time-varying machining condition data features L_T of the predicted sample at time T

The error backward transfer process consists of two stages, firstly the error is backward transferred in the GRU network, and then it continues to be backward transferred in the CNN

network. Since only convolutional layers exist in the network structure of the time-varying machining data feature extraction model in this paper, the parameter update calculation process in Step7 is given as follows:

Forward calculation process (layer $l-1$ to layer l):

$$a^{(l)} = g\left(z^{(l)}\right) = g\left(\sum_{m=1}^M z_m^{(l)}\right) = g\left(\sum_{m=1}^M a_m^{(l-1)} * w_m^{(l)}\right) \quad (9)$$

Where M is the number of convolution kernels; g is the ReLU excitation function; a_m and w_m are both network parameters.

Reverse transfer process (layer $l+1$ to layer l):

$$\delta^{(l)} = \delta^{(l+1)} \cdot \frac{\partial z^{(l+1)}}{\partial z^{(l)}} = \delta^{(l+1)} * \text{rot}180(w^{(l+1)}) e' g'(z^{(l)})$$

$$w_{new}^{(l)} = w^{(l)} + \eta \sum_{MBGD} \delta^{(l)} * \text{rot}180(a^{(l-1)}) \quad (10)$$

Where $\delta^{(l)}$ is the gradient error of the l th convolutional layer; $w_{new}^{(l)}$ is the updated weight of the l th convolutional layer; $\text{rot}180$ refers to rotating the matrix by 180° ; η refers to the learning rate, which is usually taken between $(0,1)$.

The 200×576 dimensional data matrix is obtained by the time-varying working condition data feature extraction model, that is, it contains $200 \times 1 \times 576$ dimensional vectors, where each vector corresponds to multiple time-varying working condition data within a time window of sample $C_{t,v}$. These vectors are the features of the time-varying machining working condition data after extraction.

4.3 Clamping point force prediction model construction

After the data pre-processing and feature extraction, the time-varying and non-time-varying features of the machining process were obtained. In order to improve the information perception and learning efficiency of the prediction model, we use the Concat algorithm to fuse the two types of working condition features within each sample $C_{t,v}$ and the three-way clamping force values at the clamping points to construct the input vector of the force state prediction model. Then the GRU network is used to learn the time series correlation law between the machining conditions and the force on the clamping point. The prediction model of the force state of the clamping point based on GRU network is shown in the Figure. 4 below:

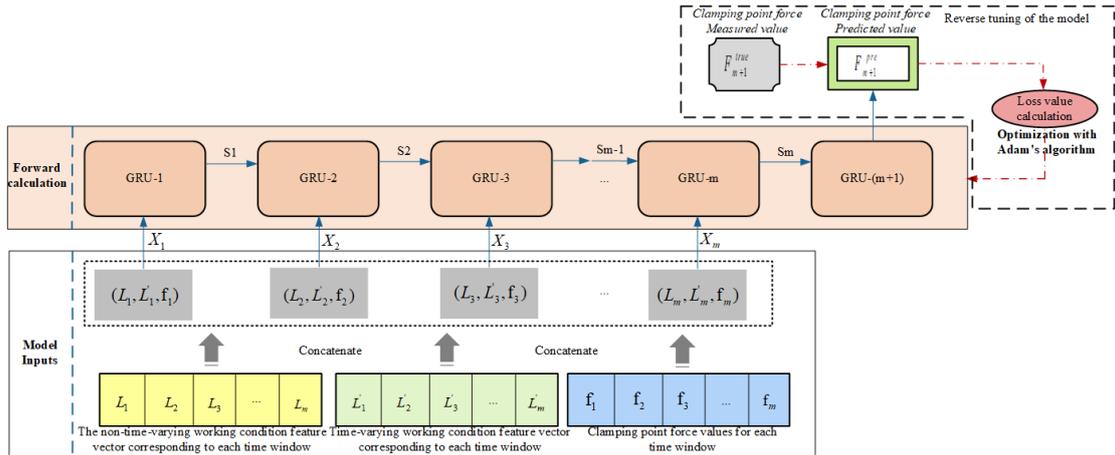


Figure. 4. Prediction model of force state of clamping point based on GRU network

(1) Model Structure

As can be seen from the above figure, the model from the bottom up mainly consists of three parts: prediction model input, prediction model forward calculation, and prediction model backward tuning. The details of each part are as follows:

The first part is the input of the prediction model. We use the Concat algorithm to fuse the non-time-varying working condition feature vector, the working condition feature vector extracted

by CNN network, and the three-way force value at the clamping point (the data value corresponding to the last moment in each time window) in each time window of the current time period, and then use them as the input of the prediction model. The fusion process using the Concat algorithm can be simply understood as the stitching of vectors, assuming that the non-time-varying condition feature vector in a time window Tw is $1 \times m$ dimensional, the time-varying condition feature vector is $1 \times n$ dimensional, and the three-way force value vector at the clamping point is $1 \times k$ dimensional, the input vector obtained after fusion is $1 \times (m+n+k)$ dimensional. The second is the forward calculation part of the prediction model, which mainly takes the fusion vectors in each time window of the current time period as input, and outputs the predicted values of the force state of the clamping point at a future time after forward calculation by the GRU network. Finally, the prediction model is backward tuned, and we use the error back propagation principle to update and adjust the weights and biases of the neurons in the network. In this paper, the mean square error (MSE) is used as the loss function in the backward tuning process, and the gradient descent process is optimized using the Adam algorithm.

(2) Model training

The structure of the GRU network neurons is shown in Figure. 5 below. The GRU network is a "two-input, two-output" structure, which uses two "gates" (update gate and reset gate) to operate on the incoming hidden layer information S_{t-1} at time $t-1$ and the input information x_t at time t . Each "gate" structure is composed of a Sigmoid excitation function and a point-by-point multiplication operation, where the update gate z_t determines the weight of the hidden layer information S_{t-1} at time $t-1$ in the candidate hidden layer information S'_t at time t , the reset gate r_t determines the weight of the candidate hidden layer information S'_t at time t in the hidden layer information S_t at time t . And the candidate hidden layer information S'_t at moment t is obtained from the hidden layer information S_{t-1} at moment $t-1$ passing through the reset gate and the input x_t at moment t through the excitation function \tanh .

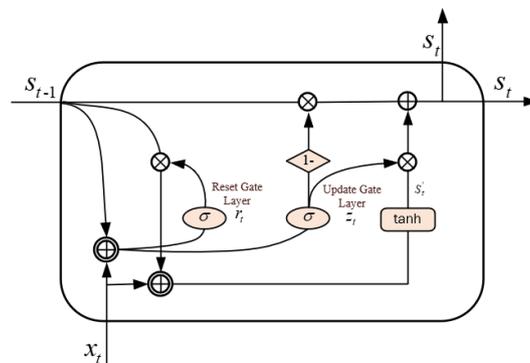


Figure. 5. The structure of GRU network neuron

The specific training procedure is as follows. The first is the forward calculation of the GRU network with the following formula:

$$\begin{aligned}
 r_t &= \sigma(W_r[S_{t-1}, x_t]) \\
 z_t &= \sigma(W_z[S_{t-1}, x_t]) \\
 S'_t &= \tanh(W_h[r_t * S_{t-1}, x_t]) \\
 S_t &= (1 - z_t) * S_{t-1} + z_t * S'_t \\
 y_t &= \sigma(W_o S_t)
 \end{aligned} \tag{11}$$

Where W_r is the weight matrix of the reset gate, W_z is the weight matrix of the update gate, W_h is the weight matrix of the hidden layer information S_{t-1} at time $t-1$ as input, W_o is the weight matrix of the output layer, $[\]$ indicates that the two matrices are connected, and y_t is the output value of the GRU network at time t .

From the forward formula, we can see that W_r , W_z , W_h , W_o are the weight parameters to be trained and learned by the GRU network, and the first three parameters can be partitioned

according to their composition as follows:

$$\begin{aligned} W_r &= [W_{rx}, W_{rs}] \\ W_z &= [W_{zx}, W_{zs}] \\ W_h &= [W_{hx}, W_{hs}] \end{aligned} \quad (12)$$

Assuming that the actual value of a sample data at moment t is y_t' and the output value at moment t obtained using the GRU network is y_t , the amount of transmission loss (with mean square error MSE as the loss function) of this network at moment t is:

$$E_t = \frac{1}{2}(y_t' - y_t)^2 \quad (13)$$

In turn, the amount of transmission loss of this network throughout the time period T is:

$$E = \sum_{t=1}^T E_t \quad (14)$$

And then the back propagation through time algorithm can be used to find the intermediate error term in the network at time t as follows:

$$\begin{aligned} \delta_{y,t} &= (y_t' - y_t) \cdot \sigma' \\ \delta_{S,t} &= \delta_{y,t} W_o + \delta_{z,t+1} W_{zs} + \delta_{t+1} W_{hs} \cdot r_{t+1} + \delta_{S,t+1} W_{rs} + \delta_{S,t+1} \cdot (1 - z_{t+1}) \\ \delta_{z,t} &= \delta_{t,S} \cdot (\hat{S}_t - S_{t-1}) \cdot \sigma' \\ \delta_t &= \delta_{S,t} \cdot z_t \cdot \tanh' \\ \delta_{r,t} &= S_{t-1} \cdot ((\delta_{S,t} \cdot z_t \cdot \tanh') W_{hs}) \cdot \sigma' \end{aligned} \quad (15)$$

The derivative of the loss quantity with respect to the weight matrix is the gradient of the weight matrix, and the summation of the gradients at each moment is the final gradient in the total time period. Therefore the gradient of each weight parameter matrix W_o , W_{zx} etc. in time period T is calculated as:

$$\begin{aligned} \frac{\partial E}{\partial W_o} &= \sum_{t=1}^T \delta_{y,t} h_t \\ \frac{\partial E}{\partial W_{zx}} &= \sum_{t=1}^T \delta_{z,t} x_t; \quad \frac{\partial E}{\partial W_{zs}} = \sum_{t=1}^T \delta_{z,t} h_{t-1} \\ \frac{\partial E}{\partial W_{hx}} &= \sum_{t=1}^T \delta_t x_t; \quad \frac{\partial E}{\partial W_{hs}} = \sum_{t=1}^T \delta_t (r_t \cdot h_{t-1}) \\ \frac{\partial E}{\partial W_{rx}} &= \sum_{t=1}^T \delta_{r,t} x_t; \quad \frac{\partial E}{\partial W_{rs}} = \sum_{t=1}^T \delta_{r,t} h_{t-1} \end{aligned} \quad (16)$$

Finally, a new round of updating of each parameter of the GRU network model can be achieved by using the above formula and selecting a suitable learning rate η .

$$\tau = \frac{\frac{m_t}{1-p^t}}{\sqrt{\frac{n_t}{1-q^t} + c}} \quad (17)$$

$$m_t = p \times m_{t-1} + (1-p) \times g_t$$

$$n_t = q \times n_{t-1} + (1-q) \times g_t^2$$

where g_t is the current gradient value of this parameter; m_t and n_t denote the first-order moment estimates and second-order moment estimates of g_t , respectively; p and q are real numbers between 0 and 1; and c is a non-zero constant.

5 Case study

5.1 Data

Considering the machining deformation problem in the actual milling process of a diesel engine piston skirt, this section collects the relevant data during the machining process of this piston skirt and verifies the proposed method by predicting the force state of its clamping point.

Tool diameter, tool edge number, spindle speed, milling depth, and feed rate are non-time-varying data during machining, which are mainly collected by manual recording methods. For time-varying conditions such as machine vibration data, clamping point force data and cutting force signals, various sensors are required to collect the corresponding data. The following Table 3 shows the types of sensors and their acquisition frequencies.

Table 3 Data acquisition sensors

Time-varying working condition signals	Sensor type	Acquisition frequency (kHz)
Machine tool vibration	BOSCH XDK	0.1
Clamping force	Xinrui XR-D7	10
Cutting force	Spike toolholder	20

Machine tool vibration data is collected by Bosch XDK sensors. By fixing it on the machine tool machining platform and pressing the data acquisition button at the start of machining and the stop button at the end, the vibration signal data for that time period is recorded and stored on the SD card. The details are shown in the Figure. 6 below.

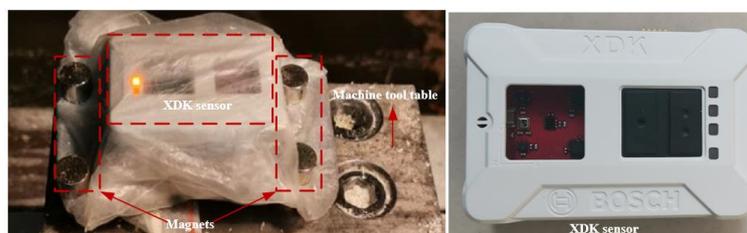


Figure 6. Machine tool vibration data acquisition

The clamping force data of the clamping point is collected by the XR-D7 three-way force sensor and the corresponding eight-channel data acquisition card. Two three-way force sensors are integrated into the fixture platen so that the sensor force point is in constant contact with the piston skirt stop without relative slippage, and then the two sensors are connected to an eight-channel data acquisition card so that the clamping force can be collected using the corresponding data acquisition software. The specific settings are shown in Figure. 7.

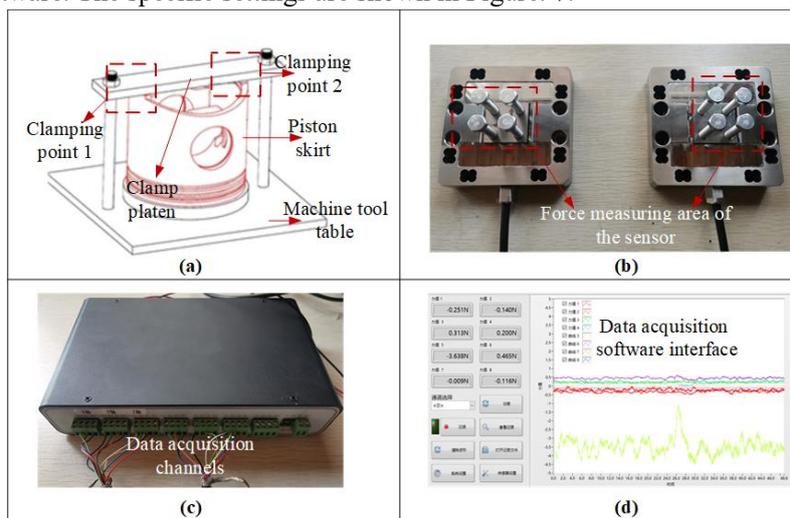


Figure 7. (a) Diagram of piston skirt clamp. (b) Three-way force sensors. (c) Eight-channel data acquisition card. (d) Data acquisition software interface.

The cutting force data is collected by means of a SIPKE toolholder. The data collected by the toolholder can be transmitted to the computer via a wireless receiver. Since the toolholder collects

the bending moment data in X/Y direction and the cutting force data in Z direction, we use the bending moment value data and the tool overhang length to calculate the cutting force data in X/Y direction. The details are shown in Figure. 8.

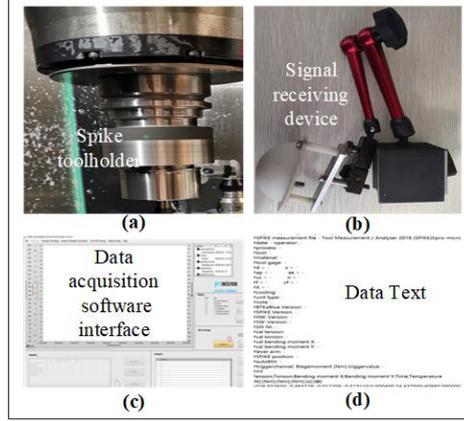


Figure. 8. (a) Spike toolholder. (b) Wireless receiving device

The milling cutter parameters are 4 edges, 8 mm diameter and 35 mm overhang. Cutting experiments were performed at 9 different cutting parameters as shown in the Table 4 below:

Table 4 Milling process parameters

Number	Spindle speed n (r/min)	Feed speed f_z (mm/r)	Cut depth a_p (mm)
1	8	220	70
2	8	250	80
3	8	280	90
4	10	220	80
5	10	250	90
6	10	280	70
7	12	220	90
8	12	250	70
9	12	280	80

5.2 Training details and comparative experiments

Firstly, the sampling interval is set for the three time-varying conditions to ensure their synchronization in terms of data volume. Then, the time synchronization of the four time-varying data is ensured by taking the time-varying data with the least amount of data as the base. The specific settings are shown in Table 5 below

Table 5 Sample interval settings

Data type	Sample interval
Machine tool vibration data	1
Clamping force data	100
Cutting force data	200

Considering that the machine vibration data acquisition frequency is only 0.1kHz, the time window T_w length and sample length are set to 0.01 seconds and 2 seconds respectively. Then, a total of 120,704 samples were obtained for nine sets of processing parameters based on the idea of rolling prediction. The input vector sequence of the clamping point force state prediction model is a sequence of 200 (sample time length/time window) time windows corresponding to the input vectors, and each input vector includes three parts, i.e., non-time-varying working condition features, time-varying working condition features, and force at the clamping point. Take the first set of cutting parameters in the table as an example to illustrate the process of obtaining the inputs to the model.

According to Table 4, the non-time-varying working condition ($C_{n,t,v}$) factors are the number of milling tool edges, tool travel mode, milling tool diameter, spindle speed, feed rate, milling depth, so the non-time-varying working condition feature vector can be denoted as follows:

$$C_{n_t_v} = (Z, w, D, n, f_z, a_p) = (4, 3, 8, 220, 70, 0.8)$$

The time-varying conditions C_{t_v} include cutting force, machine vibration, and clamping force. The three time-varying conditions are pre-processed (missing value addition, wavelet threshold noise reduction and data normalization) before feature extraction. Take the cutting force under the first set of cutting parameters as an example, set the wavelet base as db3 wavelet, the number of decomposition layers as 1, the signal length as 2, and the adjustment factor as 0.5, and then obtain the cutting force data before and after noise reduction filtering as shown in Figure. 9.

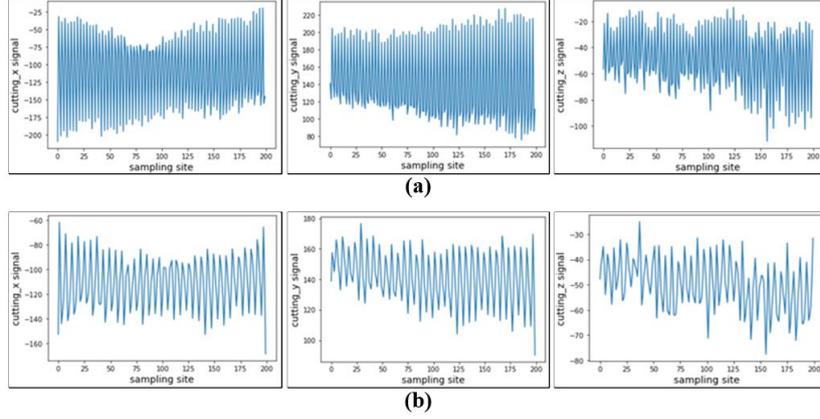


Figure. 9. (a) Cutting force data before filtering (X/Y/Z direction). (b) Cutting force data after filtering (X/Y/Z direction).

Then, the $200 \times 9 \times 1$ dimensional data matrix composed of the three time-varying conditions data after pre-processing is input into the feature extraction model proposed in Section 4.2 to obtain the time-varying condition features ($200 \times 576 \times 1$ dimensional), and then the time-varying condition feature vectors corresponding to each time window T_{w_1} to $T_{w_{200}}$ are shown below:

$$\begin{aligned} (C_{n_t_v})_{T_{w_1}} &= (-3.836, -4.662, -9.972, \dots, 0.123, -7.986, -18.148); \\ (C_{n_t_v})_{T_{w_2}} &= (-21.212, -17.560, 2.022, \dots, -5.509, 1.800, -3.569); \\ (C_{n_t_v})_{T_{w_3}} &= (-8.155, -8.168, -16.938, \dots, -9.956, 1.017, 10.735); \\ &\dots \\ (C_{n_t_v})_{T_{w_{200}}} &= (-11.954, -15.299, -15.334, \dots, -11.732, -20.324, -3.766) \end{aligned}$$

And the corresponding vector of three-way force values at the clamping point is:

$$\begin{aligned} F_1 &= (-167.544, 111.764, -4129.672) \\ F_2 &= (-141.089, -10.097, -4076.846) \\ F_3 &= (-166.400, -75.904, -4117.677) \\ &\dots \\ F_{200} &= (-27.840, 17.575, -4012.906) \end{aligned}$$

Then, the input vectors of the prediction model for each time window are obtained by fusing them using the Concat algorithm, as shown below.

$$\begin{aligned} C_1 &= (4, 3, 8, 220, 70, 0.8, -3.836, -4.662, -9.972, \dots, 0.123, \\ &\quad -7.986, -18.148, -108.981, -27.855, -4060.582) \\ C_2 &= (4, 3, 8, 220, 70, 0.8, -21.212, -17.560, 2.022, \dots, -5.509, \\ &\quad 1.800, -3.569, -43.972, -57.676, -4029.679) \\ C_3 &= (4, 3, 8, 220, 70, 0.8, -8.155, -8.168, -16.938, \dots, -9.956, \\ &\quad 1.017, 10.735, -166.400, -75.904, -4117.677) \\ &\dots \\ C_{200} &= (4, 3, 8, 220, 70, 0.8, -11.954, -15.299, -15.334, \dots, -11.732, \\ &\quad -20.324, -3.766, -27.840, 17.575, -4012.906) \end{aligned}$$

The length of the sample and the length of the time window in this paper are 2 seconds and 0.01 seconds respectively, so the constructed GRU network contains a total of 201 GRU structural units after being expanded along the time axis, where the first 200 unit structures are input into the fusion vector corresponding to 1×585 in the time window, and finally the clamping force state prediction is output in the 201st structural unit.

5.3 Result and discussion

Based on the above prediction model, we used the cross-validation method to verify the accuracy of the clamping force state prediction model. That is, eight of the nine sets in Table 6 of milling machining data sets are selected in turn as the training set and the remaining one as the test set. The cross-validation scheme is shown in the following table.

Table 6 Cross-validation schemes

Scheme No.	Training set	Test set
1	(2, 3, 4, 5, 6, 7, 8, 9)	(1)
2	(1, 3, 4, 5, 6, 7, 8, 9)	(2)
3	(1, 2, 4, 5, 6, 7, 8, 9)	(3)
4	(1, 2, 3, 5, 6, 7, 8, 9)	(4)
5	(1, 2, 3, 5, 6, 7, 8, 9)	(5)
6	(1, 2, 3, 4, 5, 7, 8, 9)	(6)
7	(1, 2, 3, 4, 5, 6, 8, 9)	(7)
8	(1, 2, 3, 4, 5, 6, 7, 9)	(8)
9	(1, 2, 3, 4, 5, 6, 7, 8)	(9)

The prediction sample data of the training set in the nine validation schemes are first randomly disrupted, and then the corresponding prediction models under the nine validation schemes are trained. The model training process is set with a learning rate of 0.0015, a batch size of 128, and a training iteration number of 200, and the gradient descent process is optimized using the Adam algorithm. Finally, the change curve of the loss function (MSE value) of the prediction model training process under the nine validation schemes is shown in Figure. 10.

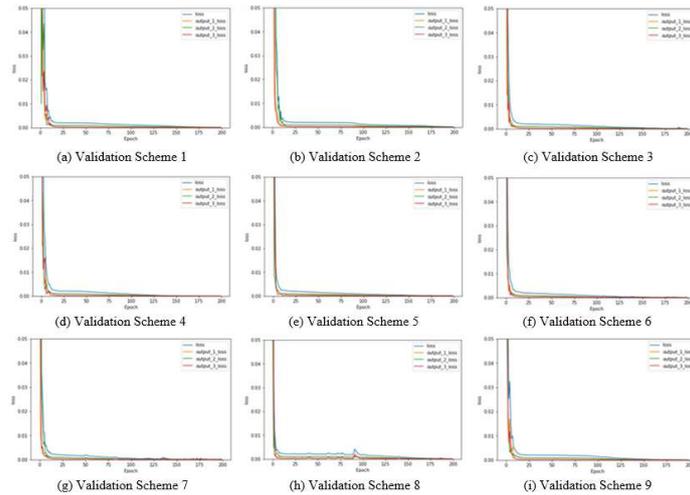


Figure. 10. The variation curves of loss values for validation schemes 1-9

The fitting effect of the predicted (X/Y/Z three-way force) curves for the test set of the nine validation schemes compared with the true value curves is shown below (only the first 100 points are shown). Prediction effects of prediction models under nine validation schemes. The results are shown in Figure. 11.

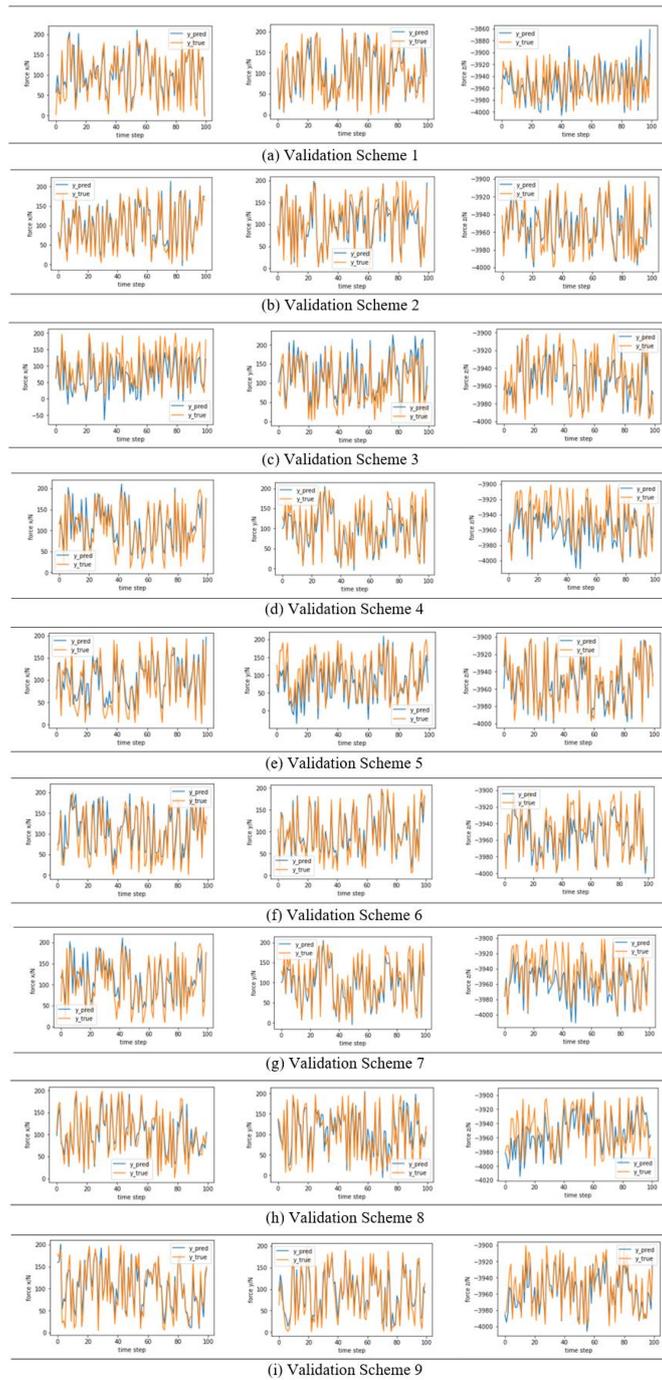


Figure. 11. The predicted results of the clamping point force prediction model under nine validation schemes
 In order to visually illustrate the prediction model effect, one sample was taken from each of the nine sets of validation scheme test sets, and the results of the true and predicted values are shown in the following Table 7.

Table 7 Comparison of predicted and real values of forces at clamping points (X/Y/Z three-way forces)

Scheme No.	Predicted value (X/Y/Z direction)	True value (X/Y/Z direction)
1	(-30.490, -12.294, -4032.832)	(-33.038, -11.664, -4036.359)
2	(-37.511, -29.436, -4049.649)	(-36.905, -28.414, -4024.089)
3	(-19.272, 45.788, -4050.098)	(-19.642, 46.682, -4051.974)
4	(-118.052, -22.212, -4077.002)	(-117.736, -23.453, -4081.122)
5	(-18.670, -31.447, -4044.574)	(-17.748, -33.085, -4038.186)
6	(-17.587, 29.659, -4103.623)	(-15.397, 33.803, -4096.761)
7	(-92.302, 106.262, -4132.234)	(-96.230, 111.247, -4140.269)
8	(-164.286, 112.488, -4053.359)	(-166.016, 111.965, -4051.934)
9	(-101.164, -29.564, -4077.147)	(-103.337, -31.856, -4150.926)

To demonstrate the advantages of the CNN-GRU prediction model, we select datasets 1 to 8 in Table 4 as the training sets, and train the CNN-LSTM model, CNN-RNN model, and CNN-BP model, respectively, while the performance of each model is evaluated in comparison using dataset 9 as the test set. Figure. 12 below shows the variation of the loss value (MSE value) for each model during the training process.

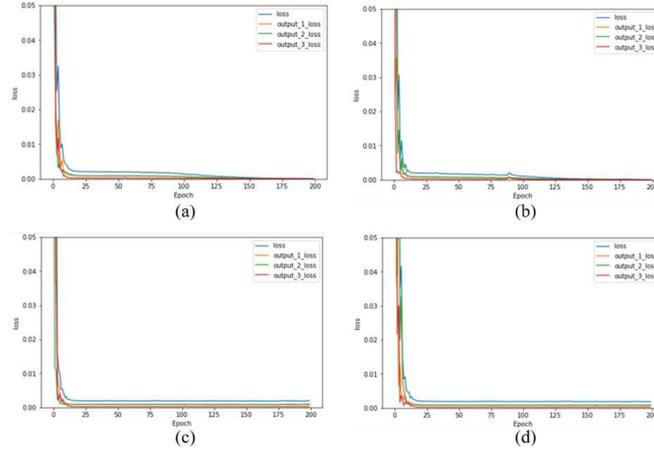


Figure. 12. (a) CNN-GRU model loss value. (b) CNN-LSTM model loss value. (c) CNN-RNN model loss value. (d) CNN-BP model loss value.

The test results of each prediction model are shown in Figure. 13 below (only the results of the three-way force prediction for clamping points with 100 samples are shown).

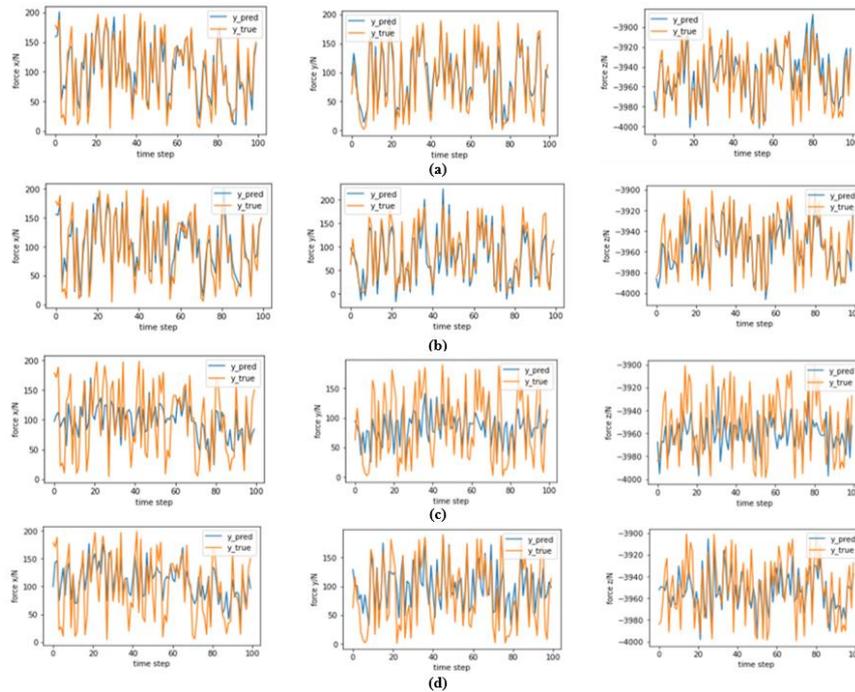


Figure. 13. (a) CNN-GRU model prediction result. (b) CNN-LSTM model prediction result. (c) CNN-RNN model prediction result. (d) CNN-BP model prediction result.

It can be seen that both the CNN-GRU model and the CNN-LSTM model have better prediction results. Further, we chose root mean square error (RMSE), mean absolute error (MAE), correlation coefficient (CC), and Nash-Sutcliffe efficiency coefficient (NSE) as assessment metrics to quantitatively compare the predictive ability of the four models. The calculated results of the mean values of the evaluation indexes for each model are shown in the following Table 8.

Table 8 Evaluation index values for the four models

	CNN-GRU	CNN-LSTM	CNN-RNN	CNN-BP
RSME	0.1869	0.2037	0.6124	0.5578
MAE	0.1568	0.1631	0.6376	0.5241
CC	0.9316	0.9187	0.7134	0.7352
NSEC	0.9435	0.9388	0.7411	0.7636

From the data in the table, it can be seen that the CNN-GRU model has the smallest RMSE and MAE values, which are 0.1869 and 0.1568, respectively; the CC and NSEC values of the CNN-GRU model are also closer to 1, which are 0.9316 and 0.9435, respectively, that is, compared with the CNN-LSTM model, CNN-RNN model, and CNN-BP model, the CNN-GRU model has better prediction ability.

6 Conclusion

Accurate prediction of the force state at the clamping point of thin-walled parts is an effective way to optimize fixture clamping and further suppress machining distortion. In this paper, we propose a CNN-GRU based force prediction model for clamping points. Our proposed method takes full advantage of the dynamic time series correlation between the complex working conditions of the machining process. The original data of the machining process is firstly pre-processed, and the CNN is used to perform adaptive feature extraction on the original training data to reduce its dimensionality while obtaining new feature data that can better reflect the essence of machining. Further, the GRU network was constructed and trained to obtain a model with the capability of retaining machining information for a long time and predicting the force on the clamping point under complex time-varying conditions, in view of the time-series correlation between the time-varying conditions and the change of the clamping point force state itself. Finally, the experimental validation was conducted using the piston skirt milling process, and the results proved the effectiveness of the CNN-GRU prediction method proposed in this paper. However, this paper only predicts the force state of key clamping points during the machining process, while the deformation of the workpiece is the result of the overall force on the workpiece. Therefore, how to extend the proposed method of force prediction at clamping points to force prediction at any point on the workpiece and the prediction of the overall force distribution of the workpiece is the future research content.

Authors' contributions Enming Li conceived of research methods, performed the experiments, and wrote the manuscript. All authors contributed to collect and analyze the data, also revisions.

Funding This work was financially supported by the National Key R&D Program of China (No.2020YFB1710305), and the Science and Technology Plan Project of Sichuan Province (No.2021YFG0052).

Availability of data and material The raw/processed data required to reproduce these findings cannot be shared at this time as the data also forms part of an ongoing study.

Declarations

Ethics approval Not applicable

Consent to participate Not applicable

Consent for publication Not applicable

Reference

1. Eduardo Diez, Hilde Perez, Vizan, Juan Marquez, Antonio Vizan (2015) Feasibility Study of In-Process Compensation of Deformations in Flexible Milling. International Journal of

Machine Tools & Manufacture , 94, 1–14.

2. Liu C, Li Y, Shen W(2018)A real time machining error compensation method based on dynamic features for cutting force induced elastic deformation in flank milling. *Machining Science and Technology*, 1-21.
3. Z Liu, M Y Wang, K Wang, X Mei (2013) Multi-objective optimization design of a fixture layout considering locator displacement and force-deformation. *International Journal of Advanced Manufacturing Technology*, 67(5-8), 1267–1279.
4. O. Gonzalo, J. M. Seara, E. Guruceta, A. Izpizua, M. Esparta, I. Zamakona, N. Uterga, A. Aranburu, J. Thoelen (2017) A method to minimize the workpiece deformation using a concept of intelligent fixture. *Robotics and Computer-Integrated Manufacturing*, 48(12), 209-218.
5. Li J G, Wang S Q (2017) Distortion caused by residual stresses in machining aeronautical aluminum alloy parts: recent advances. *The International Journal of Advanced Manufacturing Technology*, 89, 997-1012.
6. Kurnadi M S, Morehouse J, Melkote S N (2007) A workholding optimization model for turning of ring-shaped parts. *International Journal of Advanced Manufacturing Technology*, 32(7-8):656-665.
7. Dong ZH, Jiao L, Wang XB, Liang, ZQ, Liu ZB, Yi J (2016) FEA-based prediction of machined surface errors for dynamic fixture-workpiece system during milling process. *The International Journal of Advanced Manufacturing Technology*, 85, 299-315.
8. Wang JJ, Ma YL, Zhang LB, Robert X. Gao, Wu DZ (2018) Deep Learning for Smart Manufacturing: Methods and Applications. *Journal of Manufacturing Systems*, 48, 144-156.
9. Li YG, Liu CQ, Hao XZ, Gao JX, Paul G. Maropoulos (2015) Responsive fixture design using dynamic product inspection and monitoring technologies for the precision machining of large-scale aerospace parts. *CIRP Annals - Manufacturing Technology*, 64(1), 173-176.
10. Hao XZ, Li YG, Li MQ, Liu CQ (2019) A part deformation control method via active pre-deformation based on online monitoring data. *The International Journal of Advanced Manufacturing Technology*, 104(3), 2681-2692.
11. Wuest T, Weimer D, Irgens C, Thoben, KD (2016) Machine learning in manufacturing: advantages, challenges, and applications. *Production & Manufacturing Research*, 4(1), 23-45.
12. Wang MW, Zhou JT, Gao J, Li ZQ, Li EM (2020) Milling Tool Wear Prediction Method Based on Deep Learning Under Variable Working Conditions. *IEEE Access*, 8(pp. 140726-140735).
13. Pimenov DY, Bustillo A, Mikolajczyk T (2017) Artificial intelligence for automatic prediction of required surface roughness by monitoring wear on face mill teeth. *Journal of Intelligent Manufacturing*, 29(5), 1045-1061.
14. Zhao ZW, Li YG, Liu CQ, J Gao (2019) On-line part deformation prediction based on deep learning. *Journal of Intelligent Manufacturing*, 31(3), 561-574.
15. Kurnadi MS, Morehouse J, Melkote SN (2007) A workholding optimization model for turning of ring-shaped parts. *International Journal of Advanced Manufacturing Technology*, 32(7-8), 656-665.
16. Zhang FP, Yan Y, Butt SI (2016) Integrated model based thin-walled part machining precision control for the workpiece-fixture system. *International Journal of Advanced Manufacturing Technology*, 85(5-8), 1745-1758.
17. Dong ZH, Jiao L, Wang XB, Liang ZB, Liu ZB, Yi J (2016) FEA-based prediction of machined surface errors for dynamic fixture-workpiece system during milling process. *The International Journal of Advanced Manufacturing Technology*, 85(1), 299-315.
18. Simeng Liu, Xiaodong Shao, Xiao B, Ge XB, Wang D (2017) Simulation of the deformation caused by the machining cutting force on thin-walled deep cavity parts. *International Journal of Advanced Manufacturing Technology*, 92(9), 3503-3517.

19. Teramoto K (2017) On-machine estimation of workpiece deformation for thin-structured parts machining. *International Journal of Automation Technology*, 11(6), 978-983.
20. Hao, XZ, Li YG, Zhao ZW, Liu CQ (2018) Dynamic machining process planning incorporating in-process workpiece deformation data for large-size aircraft structural parts. *International Journal of Computer Integrated Manufacturing*, 32(2), 136-47.
21. Wu J, Su YH, Cheng YW, Shao XY, Deng C, Liu C (2018) Multi-sensor information fusion for remaining useful life prediction of machining tools by adaptive network based fuzzy inference system. *Applied Soft Computing*, 68, (pp.13-23).
22. Wang XQ, Zhang Y, Zhou HM, Yang FU (2016) Continuous tool wear monitoring based on hidden markov model. *Modular Machine Tool & Automatic Manufacturing Technique*, 10, 87-90. [in Chinese]
23. Zhou JT, Zhao X, Gao J (2019) Tool remaining useful life prediction method based on LSTM under variable working conditions. *International Journal of Advanced Manufacturing Technology*, 104(11), 1-12.
24. Tran D, Bourdev L, Fergus R, Torresani L, Paluri M (2014) Learning spatiotemporal features with 3d convolutional networks. 2015 IEEE International Conference on Computer Vision (ICCV) (pp. 4489-4497).
25. Elman J L (1990) Finding structure in time. *Cognitive science*, 14(2), 179-211.
26. Hochreiter S, Schmidhuber J (1997) LSTM can solve hard long time lag problems. In *Advances in neural information processing systems* (pp. 473-479).
27. Chung JY, Gulcehre C, Cho KH, BengioY (2014) Empirical evaluation of gated recurrent neural networks on sequence modeling. Eprint Arxiv, arXiv:1412.3555.
28. Li Z, Li JY, Wang Y, Wang KS (2019) A deep learning approach for anomaly detection based on SAE and LSTM in mechanical equipment. *The International Journal of Advanced Manufacturing Technology*, 103(1-4), 499-510.
29. Rui Z, Wang D, Yan R, Mao K, Wang J (2017) Machine health monitoring using local feature-based gated recurrent unit networks. *IEEE Transactions on Industrial Electronics*, 65(2), 1539-1548.
30. Zhao RM, Cui HM, Hou YL(2012) Improved threshold denoising method based on wavelet transform. *Physics Procedia*, 33, 1354-1359.
31. Xie ZJ, Song BY, Zhang Y, Zhang F (2014) Application of an Improved Wavelet Threshold Denoising Method for Vibration Signal Processing. *Advanced Materials Research*, 889–890, 799–806.
32. Zhang Weiqiang Song Guoxiang(2004) Wavelet domain signal denoising based on a new threshold function. *Journal of Xi'an University of Electronic Science and Technology*, 31(2), 296-299. [in Chinese].