Mechanical Systems and Signal Processing 193 (2023) 110241

Contents lists available at ScienceDirect

Mechanical Systems and Signal Processing

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

Improving generalisation and accuracy of on-line milling chatter detection via a novel hybrid deep convolutional neural network

Pengfei Zhang [a,b], Dong Gao [a], Dongbo Hong [b], Yong Lu [a,*], Qian Wu [a], Shusong Zan [b], Zhirong Liao [b,*]

[a] *School of Mechatronics Engineering, Harbin Institute of Technology, Harbin, China*
[b] *Faculty of Engineering, University of Nottingham, Nottingham, United Kingdom*

ARTICLE INFO

Keywords:
Chatter detection
Deep learning
Inception network
ResNet
Squeeze-and-excitation network

ABSTRACT

Unstable chatter seriously reduces the quality of machined workpiece and machining efficiency. In order to improve productivity, on-line chatter detection has attracted much interest in the past decades. Nevertheless, traditional methods are inevitably flawed due to the manually extracted features. Deep learning methods possess outstanding feature learning and classification capabilities, but the generalisation and accuracy are severely affected by the labelling and training of data. To address this, this paper proposed a novel hybrid deep convolutional neural network method combining an Inception module and a Squeeze-and-Excitation ResNet block (SR-block), namely ISR-CNN. The Inception module can automatically extract multi-scale features of cutting force signal to enrich the feature map. The SR-block can assign weights to different feature channels, thus suppressing useless feature maps and improving the model accuracy. Meanwhile, the introduction of SR-block also reduces the risk of gradient disappearance and speeds up the training of network. The generalisation and accuracy of the model is guaranteed by combining the two modules without training with transition state data. Milling tests were carried out on a wedge-shaped workpiece using different cutting parameters and tool overhang lengths to verify the accuracy and generalisability of the proposed method. The results showed that the proposed method outperforms other methods by achieving classification accuracy of on the validation and test sets 100% and 97.8%, respectively. In comparison to existing methods, the proposed method can correctly identify each machining state, including the transition states. Furthermore, the proposed method identifies the onset of chatter earlier than other methods, which is beneficial for chatter suppression.

1. Introduction

Chatter is a self-excited vibration between the workpiece and the cutting tool that occurs during milling processes. In addition to producing harmful noise, it causes machine tool damage, accelerated tool wear, decreased tool life, and poor workpiece surface quality [1–3]. Conservative cutting parameters are often chosen to eliminate chatter, which substantially reduce the productivity. To address this issue, off-line chatter prediction and on-line chatter detection are proposed. Chatter prediction is a method of selecting machining parameters prior to machining using stability lobe diagrams (SLD) [4,5]. However, the accurate acquisition of SLD is difficult because

[*] Corresponding authors.
E-mail addresses: luyong@hit.edu.cn (Y. Lu), Zhirong.Liao@Nottingham.ac.uk (Z. Liao).

https://doi.org/10.1016/j.ymssp.2023.110241
Received 8 December 2022; Received in revised form 26 January 2023; Accepted 23 February 2023
Available online 4 March 2023

Nomenclature

SLD	Stability lobe diagrams
FRF	Frequency response function
WD	Wavelet decomposition
WPD	Wavelet packet decomposition
EMD	Empirical mode decomposition
EEMD	Ensemble empirical mode decomposition
HHT	Hilbert–Huang transform
VMD	Variational mode decomposition
ANN	Artificial neural network
SVM	Support vector machine
RF	Random forest
GTB	Gradient tree boosting
ASA	Angular synchronous averaging
MFDET	Multi-feature distance evaluation technique
RFE	Recursive feature elimination
LSTM	Long short-term memory
CNN	Convolutional neural network
SE	Squeeze-and-Excitation
ResNet	Residual network
x_n^l	Feature map of the *n*-th filter in the *l*-th layer
x_m^{l-1}	The *m*-th map of the *l*-1-th layer
k_{mn}^l	Convolution kernel of the *n*-th filter in the *l*-th layer
m	Number of channels
b_n^l	Bias of the *n*-th filter in the *l*-th layer
$*$	Convolution operation
$f(\cdot)$	ReLU activation function
$\widehat{x}_i$	Output of a neuron
μ_β	Mean value of each batch β
σ_β^2	Variance of each batch β
ε	Constant that guarantees the stability of the value
N	Size of each batch
λ,α	Parameters that can be learned by the network
y_i	Output of a neuron after a BN layer
j	j-th moving step
W	Size of the pooling window
p_n^l	Feature map output from the *n*-th filter in the *l*-th layer
$x_n^{l-1}(i)$	Value of the *i*-th neuron from the *n*-th filter in the *l*-1-th layer
W^{l+1},b^{l+1}	Weight and bias of $l+1$-th layer
x^{l+1}	Output of $l+1$-th layer
$\mathbf{u}_c$	Feature map **U** with C channels
H, W	Height and width of the feature map **U**
z_c	The *c*th element of output feature map **z**
$\varphi(\cdot)$	Sigmoid activation functions
W1, **W**2	Weights of the two fully connected layers
$\mathbf{s}$	Weight coefficient of different channel feature maps
$\widetilde{x}_c$	Degree of importance of different channel feature
$\mathbf{F}_{ex}$	Two fully-connected operations
$\mathbf{F}_{scale}$	Channel-wise multiplication between the feature map $\mathbf{u}_c$ and the scalar S_c
$\mathbf{F}_{tr}$	Normal convolution operation
$\mathbf{F}_{sq}$	Global average pooling
FFT	Fast Fourier Transform
STFT	Short Time Fourier Transform
y	True label distribution
$\widehat{y}$	Predicted label distribution
i	Index of training samples in a batch
TPF	Tooth passage frequency

CF	Chatter frequency
PCA	Principal Component Analysis
NER	Normalised energy ratio
FI-VMD	Fast iterative VMD
ERD	Residual energy ratio

it depends on the frequency response function (FRF) of the cutting tool, toolholder, machine tool and workpiece material combination. Nevertheless, the FRF is usually variable and difficult to be obtained accurately during machining, which leads to limitations in SLD [6]. On-line chatter detection relies on signals collected by various sensors to identify the machining state online. Once chatter is detected, corresponding measures will be taken, such as changing cutting parameters and shutting down the machine. Moreover, chatter detection is an essential part of active chatter suppression [7–9], which usually requires chatter to be detected in a timely manner. Therefore, the chatter detection in milling operations is an essential tool for efficient and high-quality machining.

To date, a variety of chatter detection methods have been presented. It is clear that most of them involve a process of signal acquisition, feature extraction and chatter indicators design [10,11] As chatter is an unstable cut, it occurs with changes in cutting load, vibration and sound. Base on this, a variety of signals have been utilised to detect chatter, including cutting force [12–15], vibration [16–19], sound [20], current [21,22], displacement [23] and instantaneous angular velocity [24]. After the acquisition of signals, the following feature extraction is an essential step for chatter detection. Currently, different signal processing methods have been utilised to obtain sensitive chatter indicators, such as time domain method, frequency domain method and time–frequency domain method. Time domain feature extraction is relatively simple and convenient. Some dimensionless features such as kurtosis, skewness, crest factor, impulse factor, shape factor and clearance factor are usually preferred as they are less influenced by the cutting parameters [25]. Nevertheless, besides a significant amplitude change of the signal, the dominant band of the signal also shifts when chatter is present. Therefore, the Fourier transform approach is used for chatter detection. Based on the method, the chatter indicators of gravity frequency, mean square frequency and frequency variance can be extracted [26]. However, the Fourier transform method conceals the time domain information, making it difficult to locate the onset of chatter and to describe the transition period during which a non-smooth chatter signal occurs. In contrast, time–frequency domain methods such as wavelet decomposition (WD) [27], wavelet packet decomposition (WPD) [28,29], empirical mode decomposition (EMD) [30], ensemble empirical mode decomposition (EEMD) [31], Hilbert–Huang transform (HHT) [32,33], variational mode decomposition (VMD) [34,35] and multi-synchrosqueezing transform (MSST) [26] have been widely used as they provide both time and frequency information. After decomposition by time–frequency methods, chatter indicators of various energy ratio and entropy can be extracted. For example, Cao et al. [36] selected sensitive intrinsic mode functions (IMFs) based on EEMD, and extracted power spectral entropy as a chatter indicator; Zhang et al. [37] presented a fast iterative VMD to decompose the vibration signal and chose the chatter indicator of residual energy ratio; Hao et al. [38] determined the basis function and the number of layers of WPD based on the margin and the power indicator, and extracted maximum power entropy indicator of WPD. It should be noted that an explicit threshold of chatter indicator should have been designed in the above works. However, most of them either did not involve the setting of a threshold or drew a threshold by virtue of manual experience. Most chatter indicator threshold is only suitable for simple working conditions. Once the working conditions change, the threshold value may change accordingly, resulting in a non-unique threshold value. When complex operating conditions (e.g. beat frequency effects [39]) occur, the traditional threshold method may fail.

Chatter detection can be considered as a classification problem and machine learning algorithms can be competent to solve it. Traditionally, time, frequency and time–frequency domain features are extracted and filtered, and then machine learning methods such as artificial neural network (ANN) [40], support vector machine (SVM) [41], random forest (RF) [42], gradient tree boosting (GTB) [43], and so on are used for classification. Li et al. [43] acquired the chatter components by means of angular synchronous averaging (ASA) technique, then extracted the multi-scale energy entropy and built a chatter detection model using GTB. Wang et al. [44] used VMD to decompose the vibration signal, then extracted the information entropy of the decomposed signal and fed into SVM to detect robotic milling chatter. Tran et al. [45] decomposed the sound and vibration signals by WPD, then solicited time–frequency domain features and performed feature selection based on recursive feature elimination (RFE), using ANN to build a milling chatter detection model. Yesilli et al. [46] decomposed acceleration signals using WPT and EEMD, respectively, obtained time, frequency and time–frequency domain features and performed feature selection based on RFE, and finally established chatter detection model using SVM, logistic regression, RF and gradient trees, respectively. However, the works mentioned above are all shallow machine learning, requiring manual feature extraction and filtering before classification. This approach is a time-consuming and labour-intensive process. Also, it is also difficult to extract features with high generalisability, which results in extracted and filtered features that may not be suitable for variable working conditions. As a result, models built by traditional chatter detection methods are somewhat limited in their generalisability.

Deep learning, a branch of machine learning, has an extremely strong feature learning and classification capability [47,48]. Moreover, it has an end-to-end structure where all parameters can be trained together. These characteristics greatly reduce human intervention and have been widely used in the field of fault diagnosis [49,50]. Deep learning is a black box that can provide outstanding performance in representing complex non-linear physical systems, such as cutting processes that are difficult to describe with exact formulas. Several researchers have explored the use of deep learning for chatter detection. For example, Sun et al. [51] proposed a deep learning framework combining Inception module and long short-term memory (LSTM) for chatter detection considering beat frequency effects in turning process. Peng et al. [52] developed an on-line chatter detection system for milling operations using LSTM networks with the help of the motor current signal from the ball screw. Sener et al. [53] utilized the continuous wavelet transform as a pre-processing technique and then the time–frequency images were fed into a convolutional neural network (CNN) for the identification of chatter states. Altintas et al. [54] presented a combination of a machine learning network and a physics-based model for milling chatter detection, where the model built by machine learning relied precisely on CNN. It is well known that models built by deep learning require high quality data with labels. At the same time, the process usually needs to be done manually. However, most developed models are dependent on the marking of full process experimental data, which is usually based on manual judgement of the surface of the part. This approach is not only time-consuming and labour-intensive, but the labelled data can easily lead to misjudgements especially in transition states such as stable to chatter. Therefore, data labelling should be paid great attention when the chatter detection methods are established by deep learning networks. Otherwise, the model built from the wrong labels may have adverse results for chatter detection accuracy [55].

In summary, various advanced signal processing methods and machine learning algorithms have been widely used for chatter detection. Among these, traditional chatter detection methods require advanced signal processing technologies and design of sensitive chatter indicators. Besides, this method is generally difficult to determine the threshold value to accommodate multiple cutting conditions. Even though manually extracted features can be fed into machine learning algorithms for classification, it not only is time-consuming and labour-intensive, but also has certain limitations in generalisation. Compared to those, deep learning has outstanding performance in terms of feature learning and classification capabilities, which is increasingly be applied for chatter detection. However, labelling data between transition states is difficult and prone to misclassification, which affects the chatter detection accuracy. To address them, a chatter detection approach in milling process is presented based on a novel hybrid deep CNN using Inception and SR-block. The main contributions of this paper are summarised as follows.

(1) All transition state data is discarded so that the data used to construct the dataset can be easily and correctly labelled. The established deep learning model based on the pre-processing method is rare for milling chatter detection. Ultimately, it can predict milling machining states including transition states via the generalisation of the built model.
(2) To avoid manual extraction of chatter indicators and design thresholds, a novel hybrid deep CNN has been proposed using Inception module and SR-block, named ISR-CNN. In the network, the Inception module can extract multi-scale features, increase the network width, and reduce parameters and computational complexity. A Squeeze-and-Excitation block (SE block) is embedded into a ResNet block to form an SR-block, the purpose of which is to assign weights to the different feature channels. In addition, ResNet block can further extract features, and help to reduce the risk of gradient disappearance during training network.
(3) To verify the generalisation performance, the training and test sets are trained and tested with different cutting conditions data, respectively. The cutting conditions mentioned here are not only the differences in cutting parameters, but also the different lengths of the tool overhang.
(4) The paper identifies three machining states: air cut, stable and chatter. To detect chatter early, the chatter states are divided into slight and severe chatter states based on small probability assumptions.

2. Theoretical basis of model

2.1. CNn

LeCun [56] developed the CNN in 1998 with LeNet-5, a network that achieved high accuracy in handwriting recognition and became the pioneer of CNNs. Because of the shared traversal of convolutional kernels, the training parameters of CNNs are greatly reduced compared to fully connected neural networks. After more than two decades of development, CNNs are now widely used in speech recognition, image processing and other fields. The basic CNN mainly consists of convolutional layer, batch normalisation layer, pooling layer and fully connected layer [57].

2.1.1. Convolutional layer

The convolutional layer, also called the feature extraction layer, is a convolution operation of the filter kernel with the local region of the input, followed by an activation unit to output the features. The operation of the convolution layer is described as follows.

$$x_n^l = f\left(\sum_m x_m^{l-1} * k_{mn}^l + b_n^l\right) \tag{1}$$

where x_n^l represents the obtained feature map of the *n*-th filter in the *l*-th layer; x_m^{l-1} represents the *m*-th map of the *l*-1-th layer; k_{mn}^l represents the convolution kernel of the *n*-th filter in the *l*-th layer; *m* refers to the number of channels; b_n^l represents the bias; $*$ refers to the convolution operation; $f(\cdot)$ represents the ReLU activation function used in this paper.

2.1.2. Batch normalisation layer (BN layer)

Design of BN layer is to solve the problem of gradient disappearance and explosion and increases the speed of network training. Moreover, the BN layer is usually placed before the activation function and after the convolution layer. For each batchβ, the BN layer is described as follows.

$$\widehat{x}_i = \frac{x_i - \mu_\beta}{\sqrt{\sigma_\beta^2 + \varepsilon}} \tag{2}$$

$$\mu_\beta = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{3}$$

$$\sigma_\beta^2 = \frac{1}{N}\sum_{i=1}^{N} (x_i - \mu_\beta)^2 \tag{4}$$

where $\widehat{x}_i$ represents the output of a neuron; μ_β refers to the mean value of each batchβ; σ_β^2 refers to the variance of each batchβ; N represents the size of each batch; ε is an arbitrarily small constant to maintain numerical stability. To restore the expressive ability of the network, a transformation step is performed as follows.

$$y_i = \lambda \widehat{x}_i + \alpha \tag{5}$$

where λ and α are parameters that can be learned by the network in the optimization process; y_i is the output of a neuron after a BN layer.

2.1.3. Pooling layer

The pooling layer, also known as the down sampling layer of feature map, is usually placed after the convolution layer. The pooling commonly used is maximum or average pooling, which is defined as follows.

$$p_n^l = \max_{(j-1)W<i<jW}\{x_n^{l-1}(i)\} \tag{6}$$

or

$$p_n^l = \text{mean}_{(j-1)W<i<jW}\{x_n^{l-1}(i)\} \tag{7}$$

where W refers to the size of the pooling window; j is the j-th moving step; p_n^l represents the feature map output from the n-th filter in the l-th layer; $x_n^{l-1}(i)$ represents the value of the i-th neuron from the n-th filter in the l-1-th layer.

2.1.4. Fully connected layer (FC layer)

After the network has extracted features through several convolutional and pooling layers, the obtained features are connected to a FC layer. The expression for FC layer is described below.

$$x^{l+1} = f(W^{l+1}x^l + b^{l+1}) \tag{8}$$

where W^{l+1} and b^{l+1} represent the weight and bias of $l+1$-th layer respectively; x^{l+1} refers to the output of $l+1$-th layer.

2.2. Inception module

Deep CNNs can improve their performance by relying on stacking the number of network layers, such as AlexNet [58] or VGGNet [59]. However, this behaviour not only leads to an increase in the number of network parameters, but also to a risk of overfitting. Also, the increase in network parameters requires a large amount of labelled data, which is one of the biggest bottlenecks in practice, as obtaining a large number of labels is time-consuming and expensive. Furthermore, the increase in network parameters leads to a dramatic increase in computational resources, which in turn results in expensive computational costs. Therefore, the Inception-V1 module is proposed by Szegedy et al. [60]. Inside the Inception module, different scales of convolution kernels (1 × 1, 3 × 3, 5 × 5) imply different perceptual fields, which enrich the extracted features. Moreover, these features are eventually fused in the channel dimension direction and used as input for the next module. In addition, the introduction of 1 × 1 convolution kernel reduces the feature dimensionality, decreasing the size of the network parameters and reducing the computational complexity. This Inception structure allows both the width of each layer to be widened and the depth of the network to be increased without increasing the computational effort, which is beneficial for improving the generalisation and accuracy of chatter detection.

However, the standard Inception-V1 structure is suitable for processing two-dimensional data such as images. The cutting force data used in this paper is a one-dimensional time series, so the Inception-V1 structure needs to be modified and named Inception-IV1. Inspired by Inception-V3 [61], one-dimensional large convolution kernels of 5 × 1, 7 × 1 and 9 × 1 are adopted in the Inception-IV1 module due to their greater non-linear expressiveness.

2.3. SR-block

Although the deep neural networks are capable of soliciting more specific and high-dimensional features, the detailed features of input signal may also be lost as the depth of the network increases. Moreover, blindly increasing the depth of the network may lead to larger training errors, i.e. network degradation. The reason for this is that the error in back propagation may become very small as the

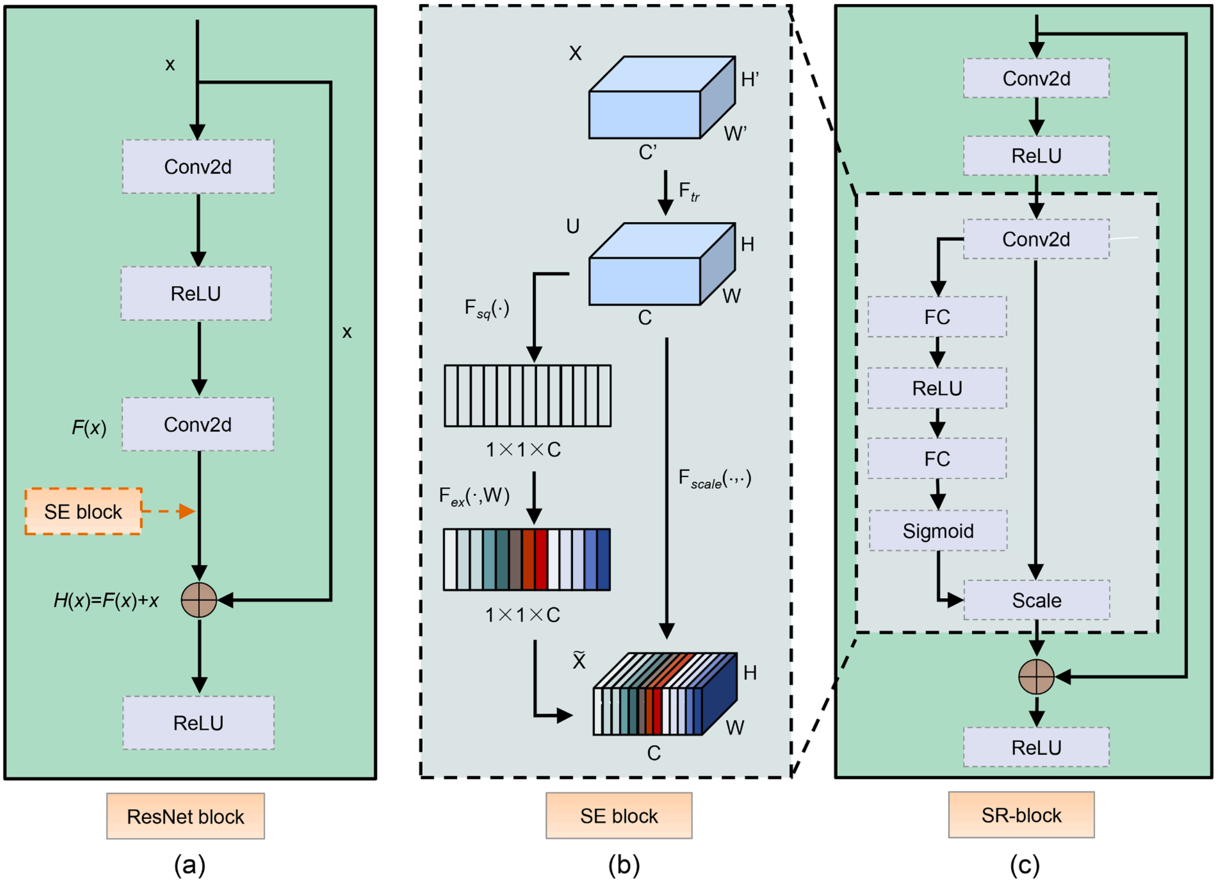

Fig. 1. Model of SR-block: (a) ResNet block; (b) SE block; (c) SR-block.

network goes deeper, which leads to a phenomenon where the weights of the intermediate layers do not change, i.e. so-called gradient disappears. To address these issues, skip connection was introduced into the network by He et al. [62] to construct a residual unit with a constant mapping, as shown in Fig. 1(a). The skip connection allows the input and output to be directly connected, which results in the gradient being directly propagated to a shallow layer, avoiding network degradation and gradient disappearance in deep networks. Assuming that $\mathbf{x}$ and $H(\mathbf{x})$ are the input and output of the residual unit respectively, and that the underlying mapping of the residual unit is: $H(\mathbf{x}) = F(\mathbf{x}) + \mathbf{x}$, the network actually learns the residual function: $F(\mathbf{x}) = H(\mathbf{x})$-$\mathbf{x}$. Moreover, the learning of the residual function has been shown to be more efficient than the original function [63].

Once the input has been feature-extracted by the Inception module and ResNet block, different features for multiple channels can be obtained. However, the importance of features for chatter detection may vary from channel to channel. A Squeeze-and-Excitation network [64] is shown in Fig. 1(b) as a channel attention mechanism. It is also named SE block which is used to improve the accuracy of chatter prediction. The SE block can establish dependencies between channels, allowing the networks to selectively boost beneficial feature channels and suppress useless ones by using global information. These characteristics enable the feature channels to be adaptively calibrated, leading to an improvement of model classification accuracy. In this study, the SE block is embedded after the second convolutional operation in the ResNet block, thus forming the SR-block, as displayed in Fig. 1(c).

As can be observed in Fig. 1(b), a feature input $\mathbf{X}$ of size $\acute{C}\times\acute{W}\times\acute{H}$ is transformed into a feature $\mathbf{U}$ of size C × W × H by the $\mathbf{F}_{tr}$ operation, where $\mathbf{F}_{tr}$ refers to the second convolution operation in the ResNet block. After the feature $\mathbf{U}$ is subjected to the operations of $\mathbf{F}_{sq}$, $\mathbf{F}_{ex}$ and $\mathbf{F}_{scale}$, the feature $\widetilde{X}$ can be obtained.

The $\mathbf{F}_{sq}$ represents a global average pooling applied to each channel of the feature $\mathbf{U}$, so that a feature map $\mathbf{z}$ of size 1 × 1 × C can be obtained. The mathematical expression for this $\mathbf{F}_{sq}$ operation is described as follows.

$$z_c = \boldsymbol{F}_{sq}(\boldsymbol{u}_c) = \frac{1}{H \times W}\sum_{i=1}^{H}\sum_{j=1}^{W} u_c(i,j) \tag{9}$$

where $\mathbf{u}_c$ represents a feature map $\mathbf{U}$ with C channels; H and W are the height and width of the feature map $\mathbf{U}$, respectively. z_c refers to the cth element of output feature map $\mathbf{z}$.

The $\mathbf{F}_{ex}$ refers to two fully-connected operations, which can feed $\mathbf{z}$ into a fully-connected layer with weight $\mathbf{W}$. A feature map of size

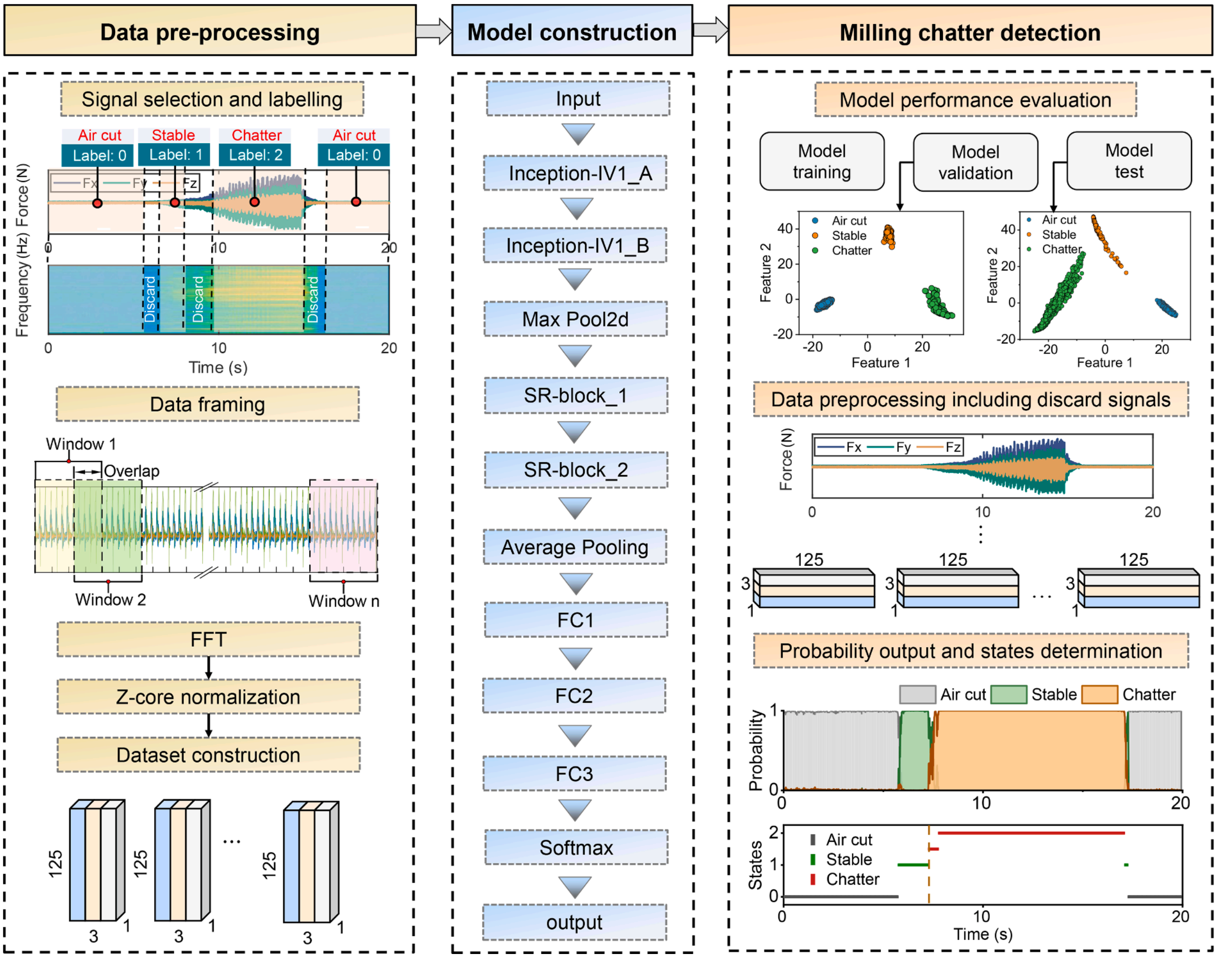

Fig. 2. Flow of proposed chatter detection method.

1 × 1 × C can be obtained as follows.

$$s = F_{ex}(z, W) = \varphi(f(z, \mathbf{W})) = \varphi(\mathbf{W}_2 f(\mathbf{W}_1 z)) \qquad 10$$

where $f(\cdot)$ and $\varphi(\cdot)$ are the ReLU and Sigmoid activation functions, respectively; **W**1 and **W**2 represent the weights of the two fully connected layers, respectively. **s** is the weight coefficient of the different channel feature maps. $\widetilde{x}_c$ represents the degree of importance of the different channel feature maps, which can be obtained by the multiplication of the weight coefficient **s** and the feature map **U** as follows.

$$\widetilde{x}_c = F_{scale}(u_c, s_c) = s_c \cdot u_c \qquad 11$$

where $\widetilde{X} = [\widetilde{x}_1, \widetilde{x}_2, \cdots \widetilde{x}_c]$ and $\mathbf{F}_{scale}$ represents the channel-wise multiplication between the feature map $\mathbf{u}_c$ and the scalar S_c.

3. The proposed chatter detection method

3.1. Flow of milling chatter detection

The proposed chatter detection method is illustrated in Fig. 2, which consists of three processes.

(1) **Data pre-processing:** The 3D cutting force signal is first selected and labelled. The transition state data is discarded as it belongs to the transition phase that is difficult to mark the state correctly. After selecting the data, the cutting force data is framed to construct the dataset. Next, the Fast Fourier Transform (FFT) is applied to the original signal as the frequency domain distribution and energy changes are most intuitively represented by the presence of chatter. The z-score normalization is then utilized to transform the input data of different amplitudes to the same magnitude, thus increasing the convergence speed of the model. Datasets including training, validation and test are constructed.

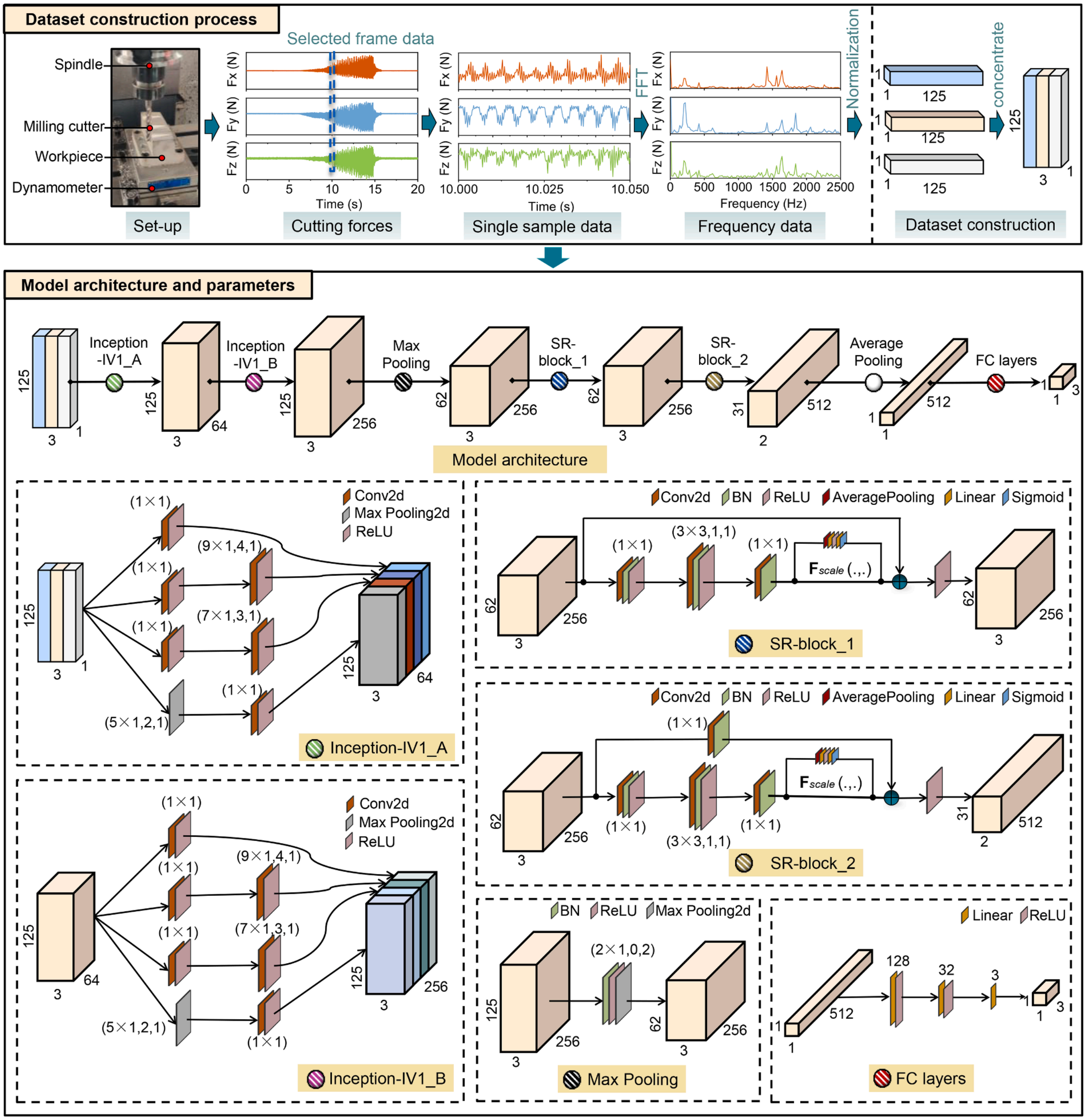

Fig. 3. Dataset and model construction of the proposed ISR-CNN.

(2) **Model construction:** A chatter detection model is constructed, which consists of two Inception modules, a maximum pooling, two SR-blocks, an average pooling and three FC layers. Finally, softmax classifier is utilised to output the probability value of states.

(3) **Milling chatter detection:** Model training, validation and test are carried out to verify the prediction accuracy and generalization ability of the model, respectively. Moreover, the test set differs from the validation set in terms of the cutting parameters and the tool overhang length during the milling process. The data containing the discarded signals for the entire cutting process is then pre-processed according to the pre-processing method described previously. Ultimately, the constructed dataset is fed into the established model for probabilistic output and states determination.

3.2. Model of the proposed ISR-CNN

Before building the ISR-CNN model, the construction process of individual datasets needs to be explained (showed in Fig. 3). In the milling experiment, 3D cutting force signals are acquired and then taken out for one frame of data. This paper uses a data size of 250 per frame, which corresponds to a sampling time of 0.05 s. Relying on the FFT and *z*-score normalization, the 3D time-domain signal is

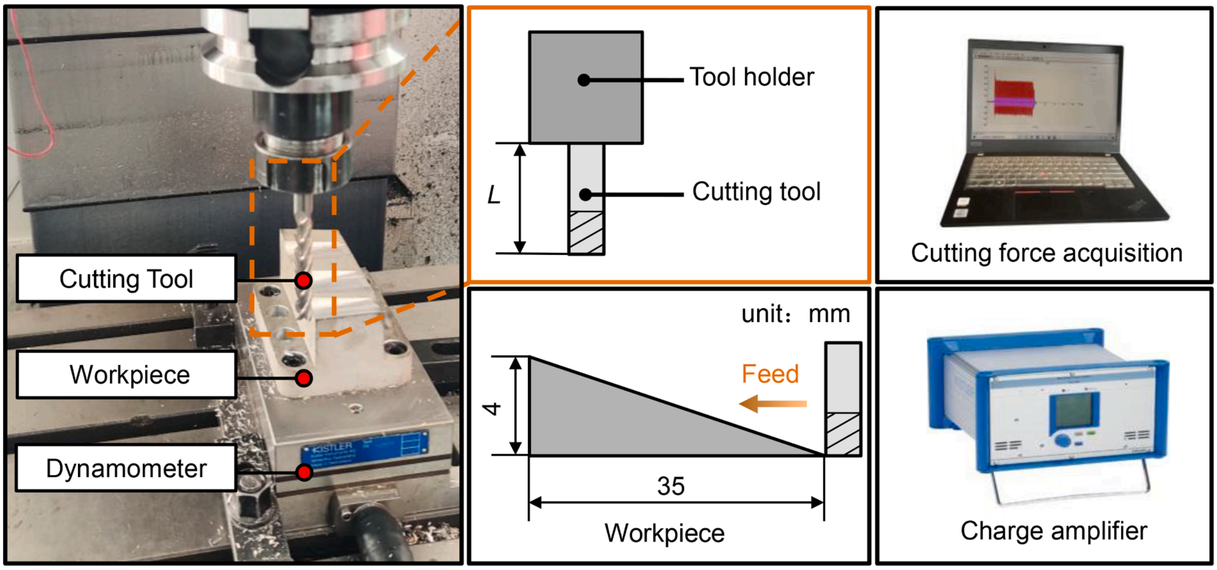

Fig. 4. Experimental setup.

Table 1
Machining parameters of milling experiments.

No.	Spindle speed n_s (r/min)	Feed per revolution f_r (mm/r)	Depth of cut a_p (mm)	Width of cut a_r (mm)	Overhang length L (mm)
1	6000	0.05	0–4	2	80
2	4800	0.05	0–4	2	80
3	4200	0.05	0–4	2	80
4	4200	0.05	0–4	1.5	90
5	6000	0.05	0–4	1.5	90

converted to the frequency domain and eventually forms a dataset of size 1 × 125 × 3.

Next, the data of size 1 × 125 × 3 is fed into the ISR-CNN model including a series of operations such as Inception module, maximum pooling, SR-block, average pooling and FC layers, as displayed in Fig. 3. After these operations, three machining states are output, i.e., air cut, stable and chatter. Inception module is capable of providing multi-scale features and has great potential in distinguishing between different machining states. In the proposed model, the Inception module is used to extract multi-scale spatial features through a series of convolutions with different kernel sizes, which is beneficial for model accuracy and generalisability. In the Inception-IV1_A module, convolution kernels of different scales (i.e., 5 × 1, 7 × 1 and 9 × 1) are used, which greatly enriches the features extracted. Ultimately, several different features are merged in the channel dimension direction. In addition, the use of 1 × 1 convolution reduces the parameters of the model and thus the complexity of the model, which is beneficial for non-expensive computations. When the input data is requested from the Inception-IV1_A module, a feature map of size 64 × 125 × 3 can be obtained and used as input to the next module. Similarly, a feature map of 256 × 125 × 3 can be obtained depending on the Inception-IV1_B. A maximum pooling module is utilized to reduce the dimensionality of input map so that a feature map of size 256 × 62 × 3 is formed. The SR-block contains a ResNet block and a SE block. The ResNet block is able to extract further detailed features and avoid gradient disappearance during network training compared to a plain CNN, which has been proved in the literature [62,65]. The SE block, namely channel attention mechanism, assigns different weights to feature channels, enabling an adaptive calibration process for features, i.e. promoting useful feature channels and suppressing useless ones, which facilitates the improvement of chatter accuracy. With the help of SR-block_1 and SR-block_2, the extracted features continue to increase in the channel dimension, resulting in a feature map of size 512 × 31 × 2. It should be noted that SR-block_2 differs from SR-block_1 in that the former uses a 1 × 1 convolutional layer in the jump junction in order to transform the shape of the input features, while the latter does not. After two SR-blocks extract features, the feature channel is finally formed. Then a global average pooling is used to obtain features in the channel dimension. Ultimately, three fully connected layers are used for dimensionality reduction, so that features of size 512 × 1 × 1 are transformed into 3 × 1 × 1, i.e. output the three machining states. In particular, the meanings of the model parameters in Fig. 3 represent the convolution kernel size, padding and stride size in that order. In addition, 1 × 1 represents the convolution kernel size and omits the padding and step size, which are 0 and 1, respectively.

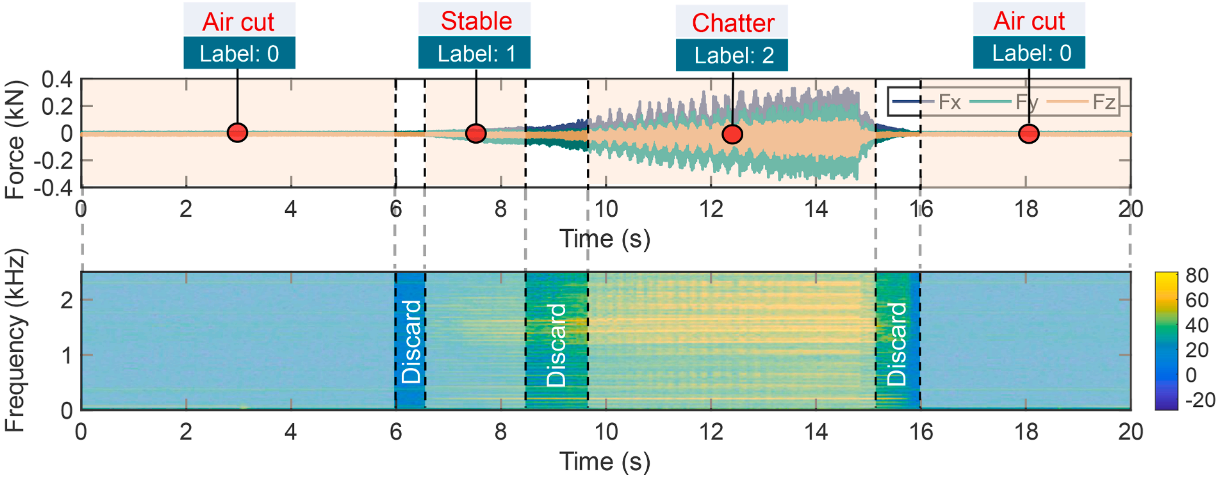

Fig. 5. Data selection and labelling.

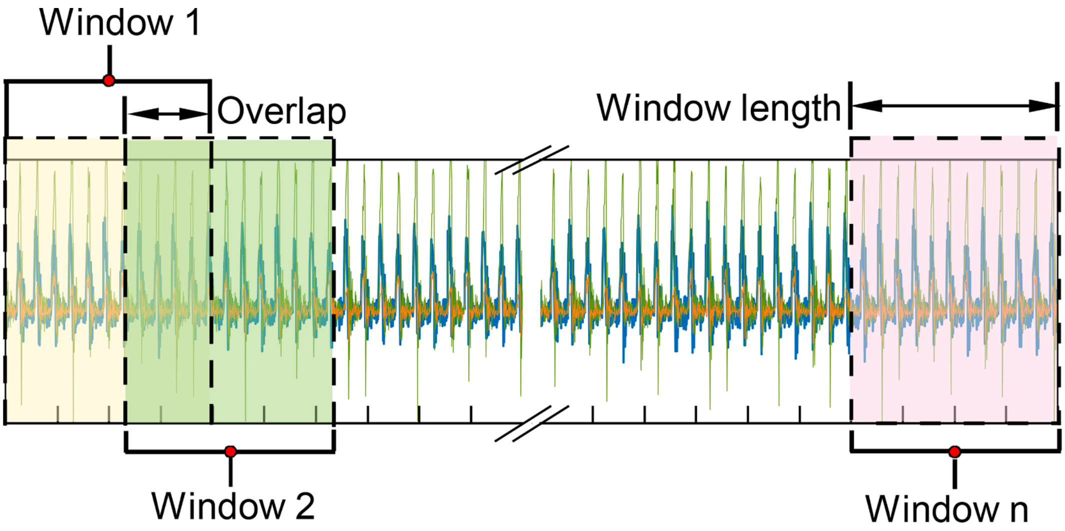

Fig. 6. Sliding time window approach.

4. Experiment and performance analysis

4.1. Experiment up and data pre-processing

To verify the effectiveness of the proposed method, milling experiments were carried out on a three- axis CNC milling machine, as shown in Fig. 4. In these milling experiments, cutting force data is used to train, validate and test the proposed model. A 10 mm diameter milling tool with 3 teeth was used in the milling process. Down milling and dry cutting are used in all cutting experiments. The cutting parameters are displayed in Table 1. It should be noted that different tool overhang lengths L are adopted in groups 1–3 and 4–5, respectively. This leads to the differences in cutting tool stiffness, which in turn affects the dynamic characteristics parameters of the cutting system. Therefore, the different lengths of L imply differences in the dynamic characteristics of the cutting system, as has been demonstrated by Yesilli et al. [46]. The wedge-shaped workpiece is made of AL7075 with a cross-section dimension of 35 × 4 mm^2. It can be mounted on a Kistler 9257B dynamometer to measure the three-component dynamic forces. The cutting force signal is amplified using a Kistler 5070 multichannel charge amplifier, which converts the charge signal into a voltage signal. An acquisition board is then utilised and the cutting force signals are collected at a sampling frequency of 5 kHz.

The collected cutting force data should be labelled before the proposed chatter detection model is built. Three machining states including air cut, stable and chatter are identified. They should be correctly classified, otherwise the training model may produce unfavourable results for the identification of machining states. However, at present, the labelling of data relies on manual empirical judgement of the workpiece surface and spectrum in the chatter detection algorithm, which can easily lead to misclassification,

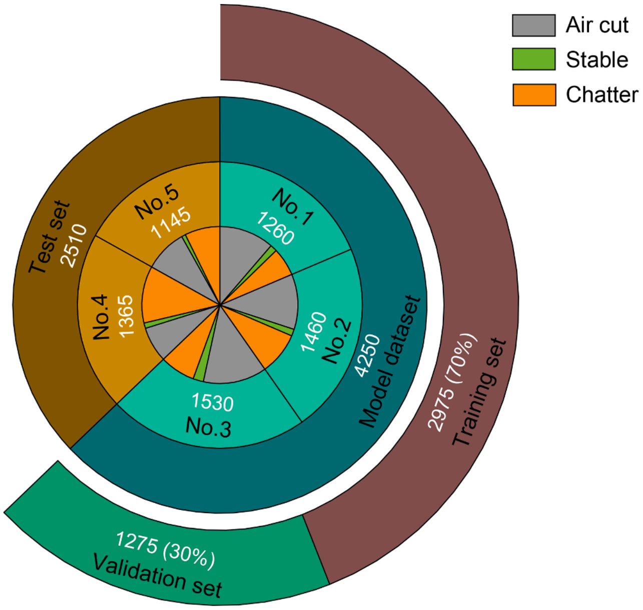

Fig. 7. Division of milling experimental dataset.

Table 2
Training and validation process of the proposed ISR-CNN for chatter detection.

```
Input: Training set $\{x_i, y_i\}_{i=1}^{s}$ and validation set $\{x_j\}_{j=1}^{t}$
Output: Classification results for machining state $\{y_j\}_{j=1}^{t}$

Initialize: Inception modual, SR-block, pooling layer and fully-connected layer parameters using random numbers
Repeat:
  Forward Propagation:
    Extraction of multiscale features from the FFT spectrum of a signal using two Inception modual
    Max pooling network is applied to reduce the dimensionality of feature map
    SR-block is used to extract detailed features, assign different weights and handle the degradation problem
    Average pooling network is applied to obtain the extracted feature
    Fully connected and softmax layers are used for machining state classification
Back Propagation:
    Compute the networks gradient $\nabla L(y_i, \hat{y}_i)$ using Adam optimizer and update network parameter
  Learning rate scheduler
Until the end of maximum iterations
Use the trained model to identify chatter online
```

especially in transition states such as air cut to stable, stable to chatter, chatter to stable and stable to air cut. As shown in Fig. 5, this study discards the data of transition states according to the Short Time Fourier Transform (STFT) spectrum so that the labels of the data can be correctly labelled. The practice avoids the difficulty of labelling the transition states. Although the trained model lacks data on transition states, the established model can still be applied for determination of them, which mainly relies on the deep learning algorithm to extract more high-level features with better generalisation. In addition, for the purpose of model training, validation and test, the data for the air cut, stable and chatter states are labelled as 0, 1 and 2 respectively in this paper.

The cutting force are framed to construct the dataset after selecting and labelling. Here the sliding time window approach is used as shown in Fig. 6. Considering the real-time performance of on-line chatter identification, the length of each window is set at 250, which corresponds to a sampling time of 0.05 s. Between adjacent windows, the movement of the previous window with respect to the latter window is 50, corresponding to an overlap of 200, which represents an update of the results every 0.01 s for on-line chatter detection.

Following the method above, the transition state data was discarded, and data from Nos. 1–5 was selected and labelled for the air cut, stable and chatter states, respectively. Based on the sliding time window method in Fig. 6, datasets for each experiment were constructed. After the FFT and z-score normalization were performed, the final datasets were formed as displayed in Fig. 7. Nos. 1–3 constitute the model dataset for training and validating the prediction accuracy of the model while nos. 4 and 5 constitute the test set for testing the generalisation performance of the model. Moreover, the model dataset is randomly divided into a training set and a validation set in the ratio of 7:3. Therefore, the test set and training set is derived from different experiments, which leads to differences in cutting parameters and tool overhang lengths. The sizes of these datasets are depicted in Fig. 7.

Table 3
Hyperparameters for training.

Hyperparameters	Values
Batch size	32
Learning rate	0.0001
Weight decay	0.001
Epoch	50
Learning rate scheduler	StepLR
StepLR.step_size	10
StepLR.gamma	0.8

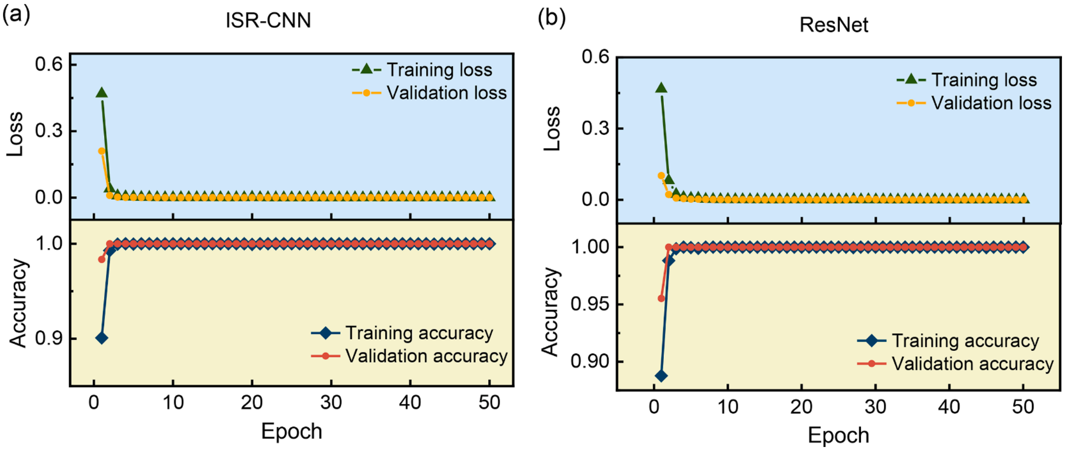

Fig. 8. Loss and accuracy during training and validation processes: (a) ISR-CNN;(b) ResNet.

4.2. Model parameter setting and training

The loss function and hyperparameters should be defined before the model can be trained. Using the cross-entropy loss as the loss function, the model can be trained in an end-to-end manner based on a backward propagation algorithm. Let us define $\mathbf{y}$ and $\widehat{\mathbf{y}}$ to denote the true label distribution and the predicted label distribution, respectively. The goal of training is to minimise the cross-entropy loss between $\mathbf{y}$ and $\widehat{\mathbf{y}}$ for all training samples. The mathematical expression for the cross-entropy loss is described as follows.

$$L(\mathbf{y},\widehat{\mathbf{y}}) = -\sum_{i=1}^{s} y_i \log(\widehat{y}_i) \quad 12$$

where s is the number of training samples; i is the index of training samples.

The training and validation process is displayed in Table 2. During the training process the parameters of each layer of the network are initialised randomly, but meet a specific distribution to better update the weights and bias parameters. In this study, we utilise the pytorch package to build the deep learning network. As observed in Table 2, the Adam optimizer was applied to complete the back propagation of gradient descent and thus update the weight parameters.

Table 3 gives the settings of the hyperparameters during the model training process. It can be seen from Table 3, each batch size was 32 and the learning rate was set to 0.0001. This learning rate was updated every 10 steps using StepLR method until the model training stopped at 50 epochs.

To better demonstrate the effectiveness of the proposed method, two popular machine learning methods were compared. Both methods have been widely used in the fields of chatter detection, tool wear detection and fault diagnosis [12,65–67]. One of them is a deep residual network (ResNet) consisting of a stack of three convolutional layers, a maximum pooling layer, two residual blocks, an average pooling and three fully connected layers. Its model structure and parameters are showed in Appendix A. This network uses the same dataset as the proposed method, with direct input of frequency domain data. The other is to use WPD to decompose all the time domain signal containing transition states and extract the energy entropy features of the sub-signal, which is then fed into the SVM for state classification. Therefore, both the ISR-CNN and ResNet models use the same dataset of discarded transition states, while the WPD-SVM model extracts all data whose features contain transition states.

The proposed model and ResNet were trained separately, and the loss and accuracy of their training and validation process are respectively displayed in Fig. 8(a)-(b). It can be observed from Fig. 8 that as the number of epochs increases, both the training loss and the validation loss decrease rapidly in the first few steps and remain almost constant afterwards. Besides, the training accuracy and validation accuracy reaches 100 % during the training process, and the error is reduced to a minimum, which indicates that the ISR-

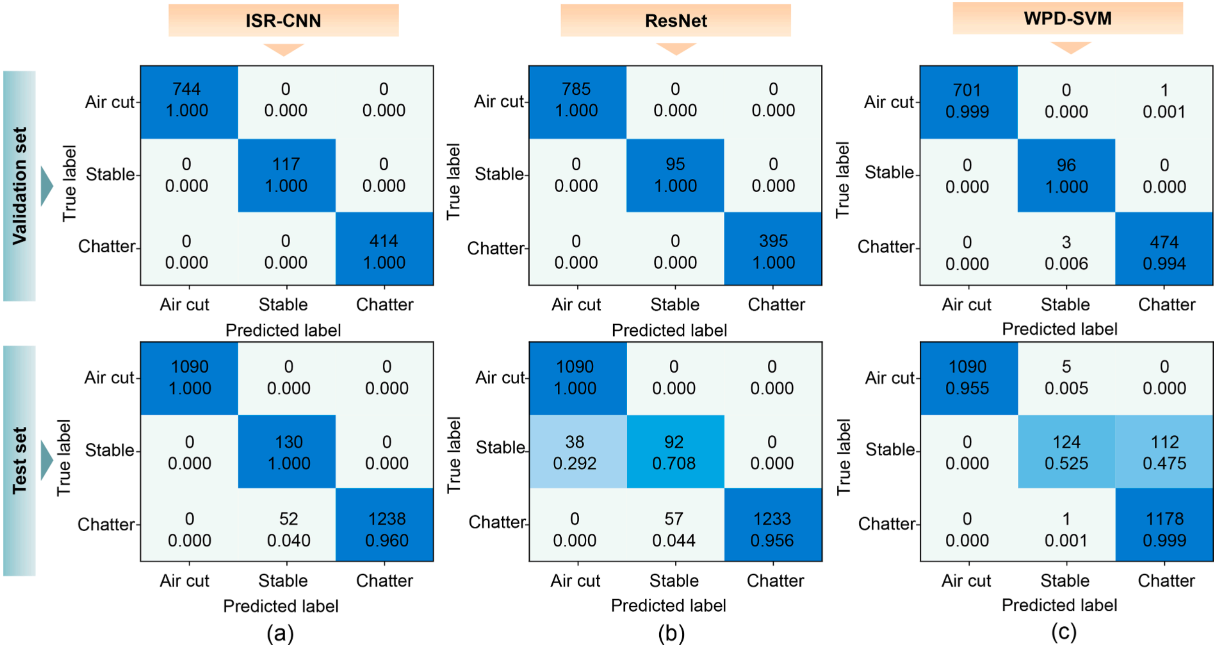

Fig. 9. Confusion matrices on validation and test set: (a) ISR-CNN; (b) ResNet; (c) WPD-SVM.

Table 4
Performance metrics of validation set.

Methods	Precision			Recall			F1 score			Accuracy
	Air cut	Stable	Chatter	Air cut	Stable	Chatter	Air cut	Stable	Chatter	
ISR-CNN	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
ResNet	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
WPD-SVM	1.000	0.969	0.997	0.999	1.000	0.994	0.999	0.984	0.995	0.996

Table 5
Performance metrics of test set.

Methods	Precision			Recall			F1 score			Accuracy
	Air cut	Stable	Chatter	Air cut	Stable	Chatter	Air cut	Stable	Chatter	
ISR-CNN	1.000	0.715	1.000	1.000	1.000	0.960	1.000	0.834	0.980	0.979
ResNet	0.966	0.617	1.000	1.000	0.708	0.956	0.983	0.659	0.977	0.962
WPD-SVM	1.000	0.953	0.913	0.955	0.525	0.999	0.976	0.677	0.954	0.953

CNN and the ResNet model used do not show gradient disappearance during the training process. At the same time, the training and validation accuracy reaches 100 % with minimal fluctuation afterwards, which also reflects the good robustness of the models built and the quality of the dataset is high after the transition state is discarded.

4.3. Model performance evaluation

After these models are trained, the confusion matrix for the validation and test sets can be obtained as shown in Fig. 9. From the validation set, it can be seen that the proposed ISR-CNN and ResNet achieve 100 % classification accuracy for each category while the classification accuracy of the other two states obtained by WPD-SVM is less than 100 % except for the stable state classification. These results demonstrate that the proposed ISR-CNN and ResNet have higher classification accuracy than WPD-SVM. The proposed ISR-CNN has a classification accuracy of 96 % in the chatter state and 100 % in the remaining two states from the test set, which are all higher than the classification accuracy of ResNet. Moreover, the recognition rate of stable state obtained by ResNet is only 70.8 %, which led to difficulties in distinguishing between the air cut and stable state. Even though WPD-SVM's discrimination in the chatter state is slightly higher than ISR-CNN, the classification accuracy in the stable state is only 52.5 %. This classification accuracy is unacceptable, as it is much less than the accuracy of the proposed method. These analyses indicate that the proposed method has higher classification

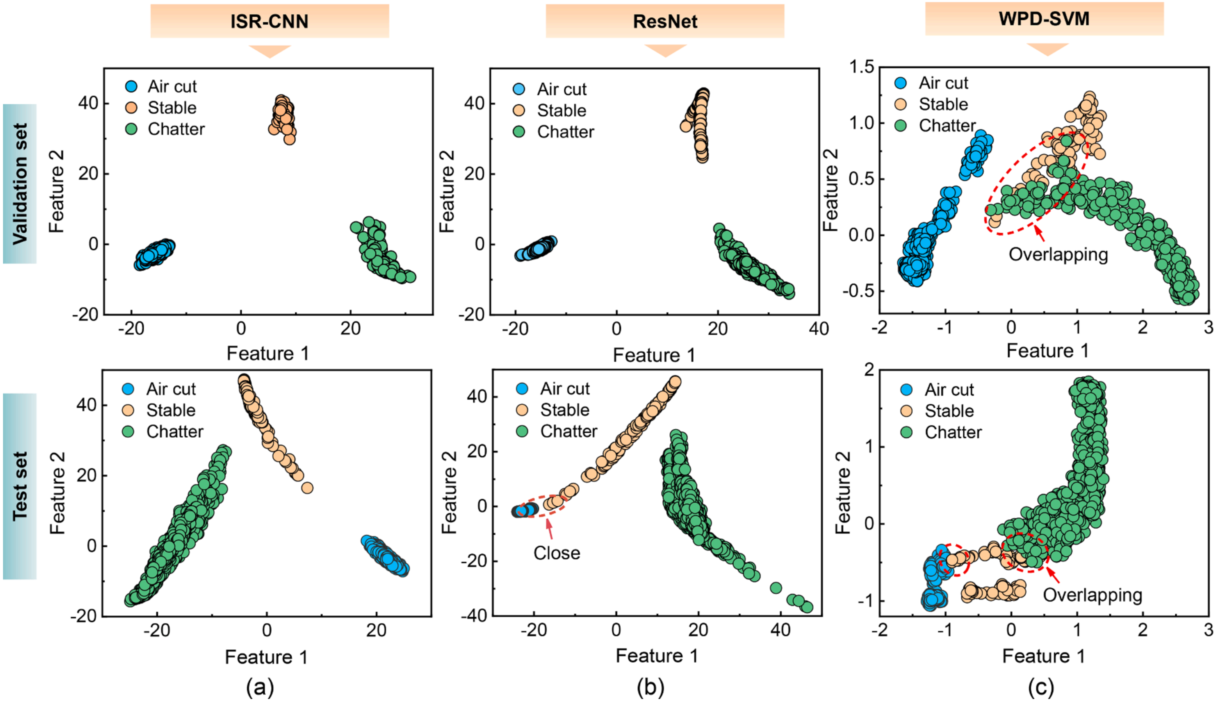

Fig. 10. Feature visualization in the last FC layer of validation and test set: (a) ISR-CNN; (b) ResNet; (c) WPD-SVM.

accuracy on both the validation and test sets. Further, the classification accuracy of each model in the test set is less than that in the validation set due to the fact that the machining parameters in the test set are different from that in the training set while the machining parameters in the validation set are the same as that in the training set.

Based on the confusion matrix in Fig. 9, the performance metrics including precision, recall, F1 score and accuracy for the validation and test sets can be calculated as shown in Tables 4 and 5 respectively. The number of data for different machining states is inconsistent, and the stable state has the least number of data. Therefore, the F1 score is introduced to balance the precision and recall. It can be seen from Table 4 that the F1 scores obtained by the proposed ISR-CNN and ResNet are higher than those obtained by the WPD-SVM regardless of the machining state. The chatter detection accuracies in the validation set are 100 %, 100 % and 99.6 %, corresponding to those obtained with the ISR-CNN, ResNet and WPD-SVM, respectively. It can be observed from Table 5 that the F1 scores obtained by ISR-CNN for each state are higher than those obtained by the other two methods. Moreover, the proposed ISR-CNN achieves the highest classification accuracy of 97.9 % on the test set. Both F1 score and accuracy demonstrate that the proposed method outperforms ResNet and WPD-SVM on the test set.

To understand the features learned by these models, Principal Component Analysis (PCA) technique is used to visualise the distribution of features in the last fully connected layer of the network. The results of PCA visualisation on the validation and test set are displayed in Fig. 10. On the validation set, there is no overlap between the clusters obtained by ISR-CNN and ResNet. The clusters are internally concentrated, and the distance between cluster centres is far apart. However, there is a partial overlap between the two clusters obtained by WPD-SVM for stable and chatter. These characteristics may lead to a lower classification accuracy in the validation set than that obtained by ISR-CNN and ResNet, which is validated by the classification accuracy of the model on the validation set in Fig. 9. On the test set, the individual clusters obtained by the proposed ISR-CNN do not overlap. Although the clusters obtained by ResNet also do not overlap, the distance between the air cut and stable clusters is very close. This feature leads to the difficulty in distinguishing between the air cut and stable, which is consistent with the results obtained by ResNet in the confusion matrix of the test set. There are some overlaps between the clusters (air cut and stable, stable and chatter) obtained by the WPD-SVM. The stable and chatter clusters fall into a severe overlap that makes them difficult to identify. This result is in line with the fact that the classification accuracy of the WPD-SVM in the test set for the stable state is only 52.5 %. In addition, the three clusters obtained by the individual models on the validation set are more concentrated and the cluster centres are more widely spaced than those obtained on the test set. These characteristics may have led to higher classification accuracies for the individual models on the validation set than on the test set.

Although the built model has achieved excellent classification accuracy on the validation and test sets, the machining state determination on the whole milling experiment is yet to be validated. This is because the transition state data for milling experiments is discarded when building the dataset. The discrimination performance of the proposed ISR-CNN, ResNet and WPD-SVM are verified separately using No. 2 as an example.

The milling experiment results for No. 2 are shown in Fig. 11. Fig. 11(a) represents the cutting forces collected throughout the

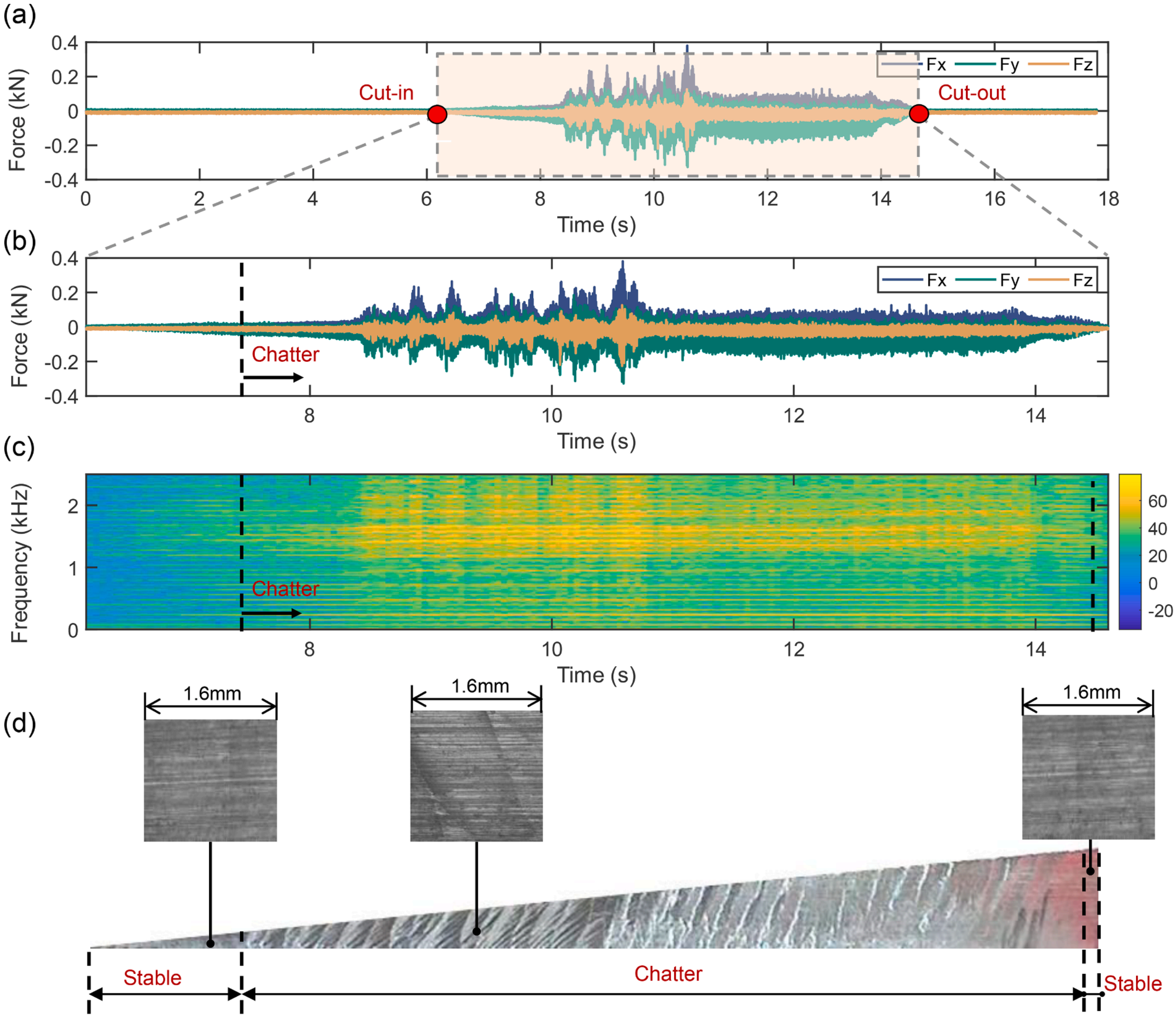

Fig. 11. Milling experiment results for No. 2: (a) cutting force collection; (b) cutting force excluding air cut; (c) STFT diagram excluding air cut; (d) surface quality of machined workpieces.

experiment. According to the FFT spectrum, the cutting forces excluding the air cut state are taken out in Fig. 11(b). Then they are subjected to STFT into a time–frequency diagram in Fig. 11(c). The surface quality of machined workpiece is displayed in Fig. 11(d). Furthermore, a white light interferometer (Zygo New ViewTM8200) was used to photograph the surface of the workpiece to determine the moment of chatter. Based on the time–frequency diagram and the workpiece surface quality, the onset and end time of chatter were determined to be 7.2 s and 14.5 s respectively.

The probabilities of each state output by the ISR-CNN, ResNet and WPD-SVM for No. 2 are depicted in Fig. 12. From Fig. 12(a), it can be observed that the probability of the proposed ISR-CNN output is close to 100 % when the machining states are located in the air cut, stable and chatter states. This characteristic leads to the three machining states being easily distinguished. When the transition state is at the transition from air cut to stable, the probability of the former state decreases continuously close to 0 and the probability of the latter state increases continuously close to 100 %. The characteristic is also true for the other transition states. As can be seen from Fig. 12(b), when the machining state is in air cut state, the probabilities of ResNet output fluctuate, i.e. inside the dashed box. Moreover, the probabilities of air cut and stable state overlap in amplitude, which makes them difficult to be distinguished. In contrast, in Fig. 12(c), when the machining state is in the chatter state, the probabilities of the WPD-SVM output fluctuate, i.e. inside the dashed box. It is difficult to distinguish between the stable and chatter when their probabilities overlap. Based on the above analysis, the proposed ISR-CNN is easier to distinguish the three states of air cut, stable and chatter than ResNet and WPD-SVM.

The usual principle for determining the machining state with the help of Softmax output is that the machining state with maximum output probability is considered to be the read cutting state. This behaviour is feasible for harmless machining states, such as air cut and stable. This is because the delay caused by this practice is usually in the millisecond range, which has little effect on harmless machining states. However, early identification is necessary for harmful machining state, such as chatter. This practice is of

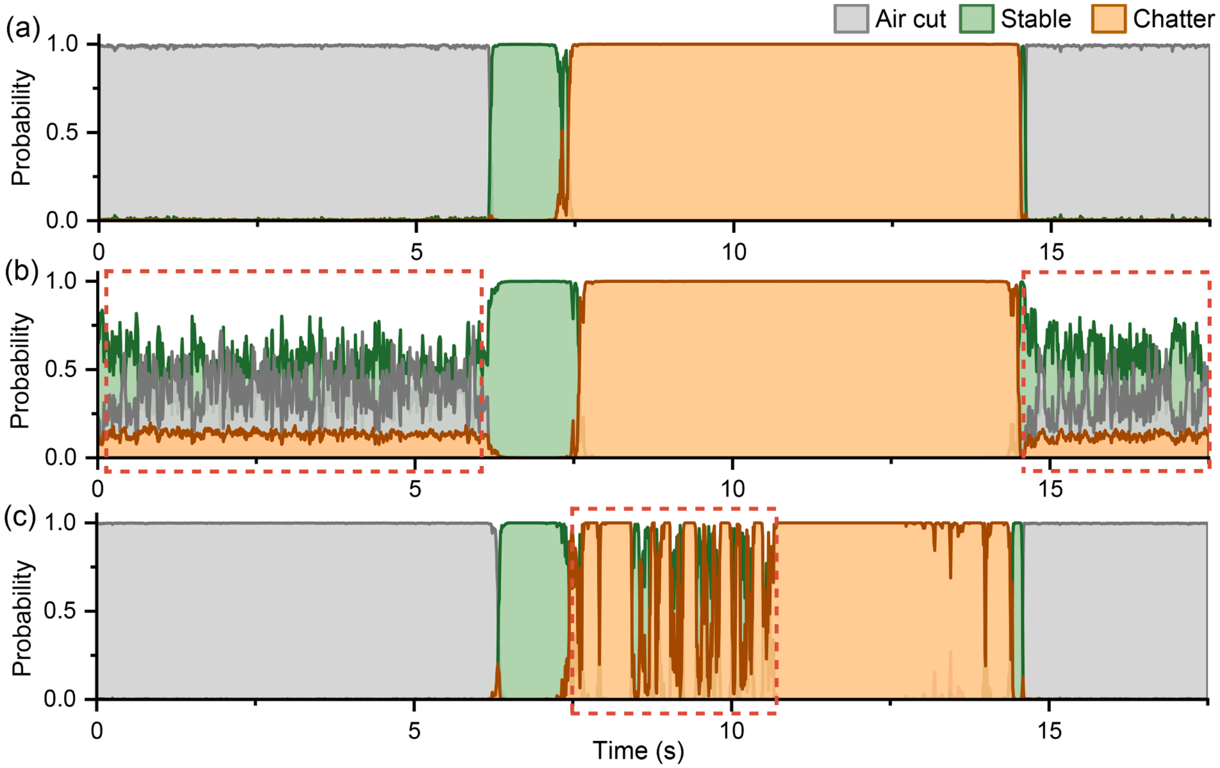

Fig. 12. Probability of each state output for No. 2: (a) ISR-CNN; (b) ResNet; (c) WPD-SVM.

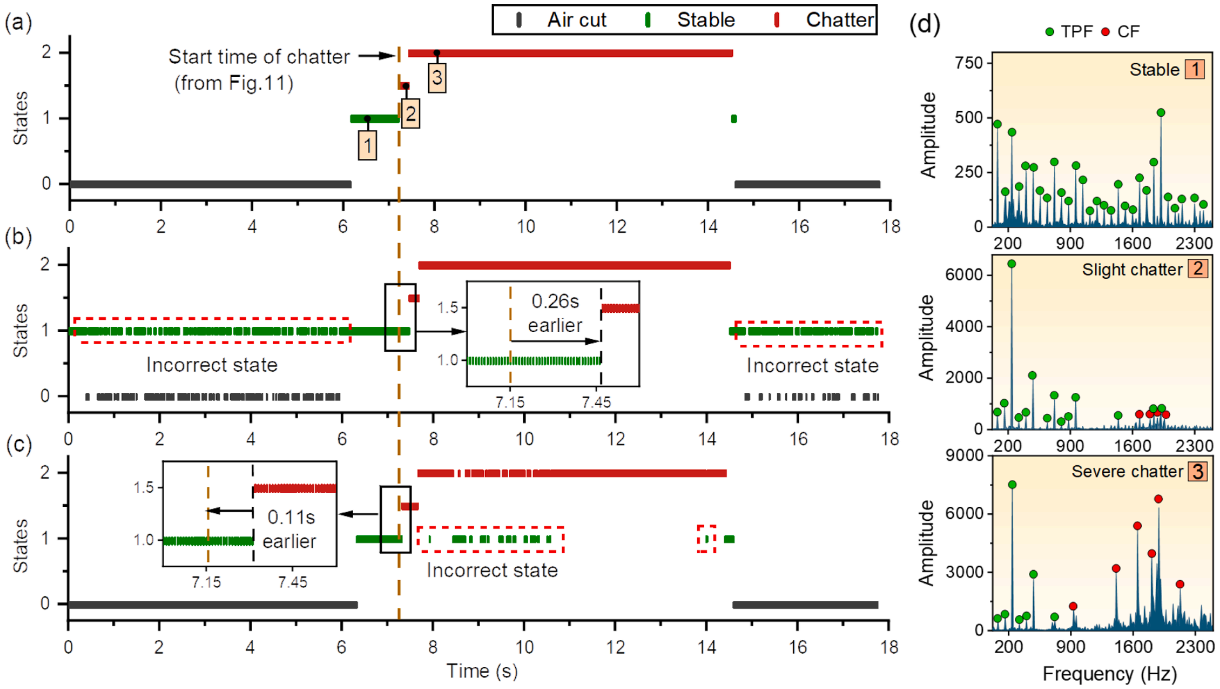

Fig. 13. States determination for No. 2: (a) ISR-CNN; (b) ResNet; (c) WPD-SVM; (d) FFT validation.

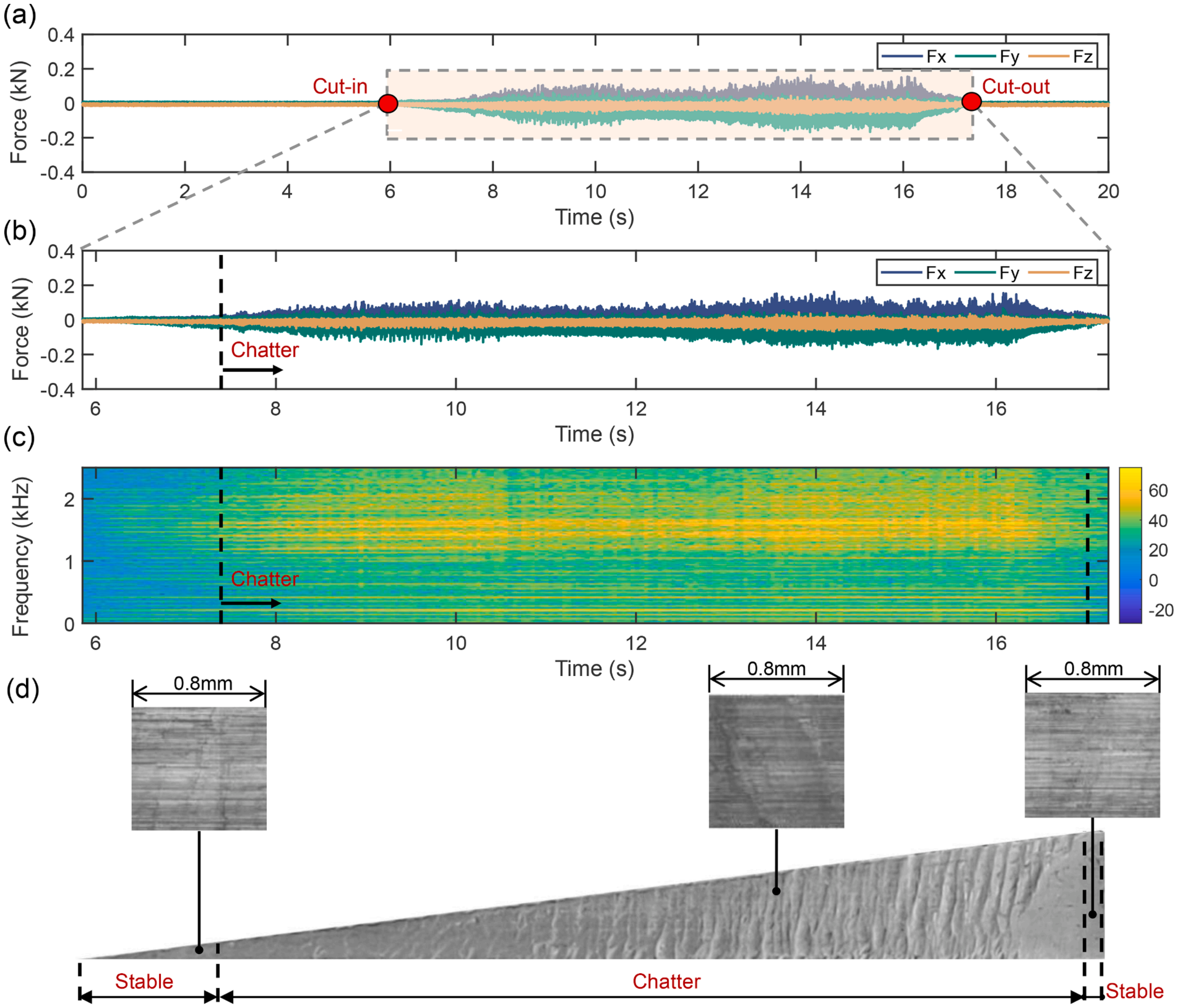

Fig. 14. Milling experiment results for No. 4: (a) cutting force collection; (b) cutting force excluding air cut; (c) STFT diagram excluding air cut; (d) surface quality of machined workpieces.

considerable importance for subsequent chatter suppression measures. Therefore, small probability assumption approach presented in the literature [68] was applied when determining the transition from stable to chatter. The principle of this approach is as follows: the appropriate criterion for state identification is the 0.05–0.95 rule based on the small probability assumption. If the probability of chatter state remains increasing until it exceeds 0.05, the small probability hypothesis will be rejected and the identification will be determined by a shift in the machining state from stable to slight chatter. If the probability of chatter states remains increasing and becomes the majority, i.e. slight chatter develops into severe chatter. Based on the above principles, the machining state of No. 2 can be determined by ISR-CNN, ResNet and WPD-SVM, as shown in Fig. 13.

It can be observed from Fig. 13(a) that the proposed ISR-CNN can detect each machining state well and the onset time to detect chatter is 7.21 s, which is in agreement with the chatter onset time determined in Fig. 11. The proposed method can identify the states from stable to chatter and from chatter to stable again. Also, the transition time from chatter to stable state is short from Fig. 12(a), but the proposed method can still identify them in Fig. 13(a), which shows the good identification ability of this method even when the transition state is ambiguous (i.e. the stable and chatter states switch repeatedly). As can be seen from Fig. 13(b), the air cut state that relies on the ResNet determination is misclassified as stable state. The time to detect the onset of chatter is 7.47 s, which is 0.26 s later than the proposed ISR-CNN detection. In contrast, in Fig. 13(c), the chatter state is misclassified as stable state based on WPD-SVM. In addition, the time to detect the onset of chatter is 7.32 s, which is 0.11 s later than that obtained by the proposed ISR-CNN. Since the ISR-CNN and ResNet methods use the same dataset of discarded transition states, the WPD-SVM method uses all the data. However, the three methods identify transition states at almost the same moment from the above analysis, which proves that the ISR-CNN and ResNet trained by discarded transition state data is credible. To verify the reasonableness of the classification of stable, slight and severe chatter, their spectra are depicted separately in Fig. 13(d). As can be observed from Fig. 13(d), when the cut is stable, the

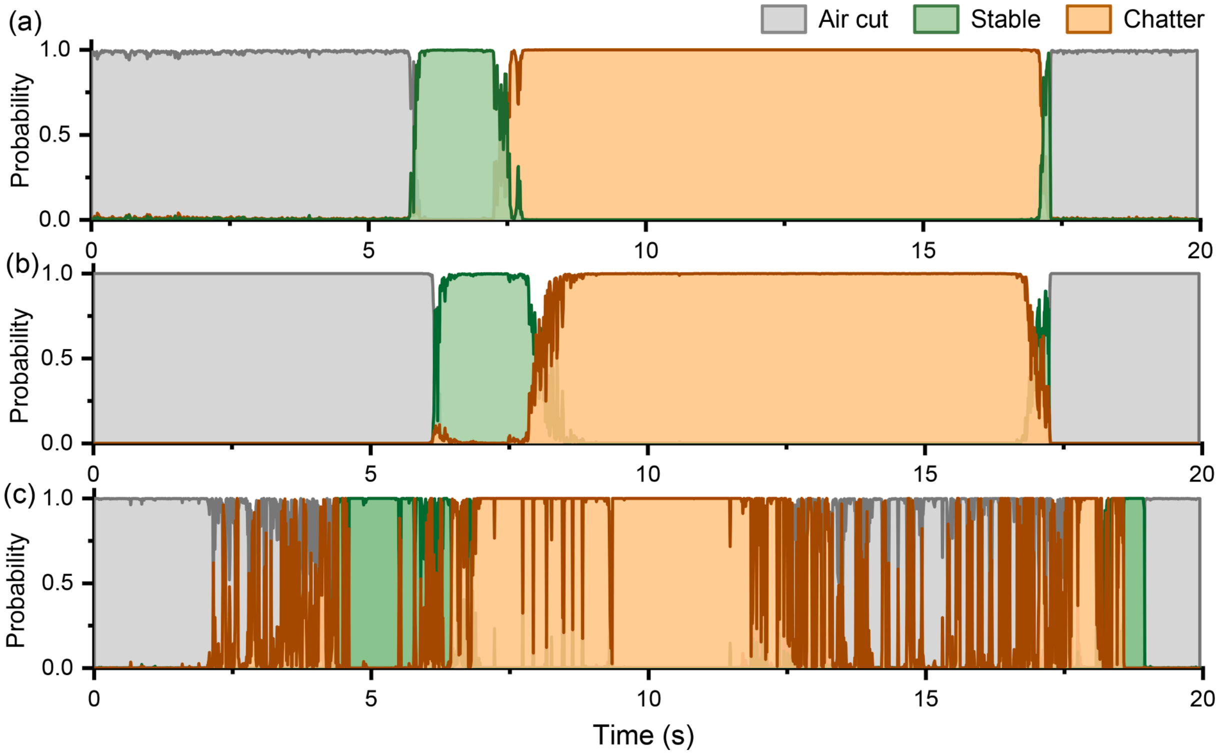

Fig. 15. Probability of each state output for No. 4: (a) ISR-CNN; (b) ResNet; (c) WPD-SVM.

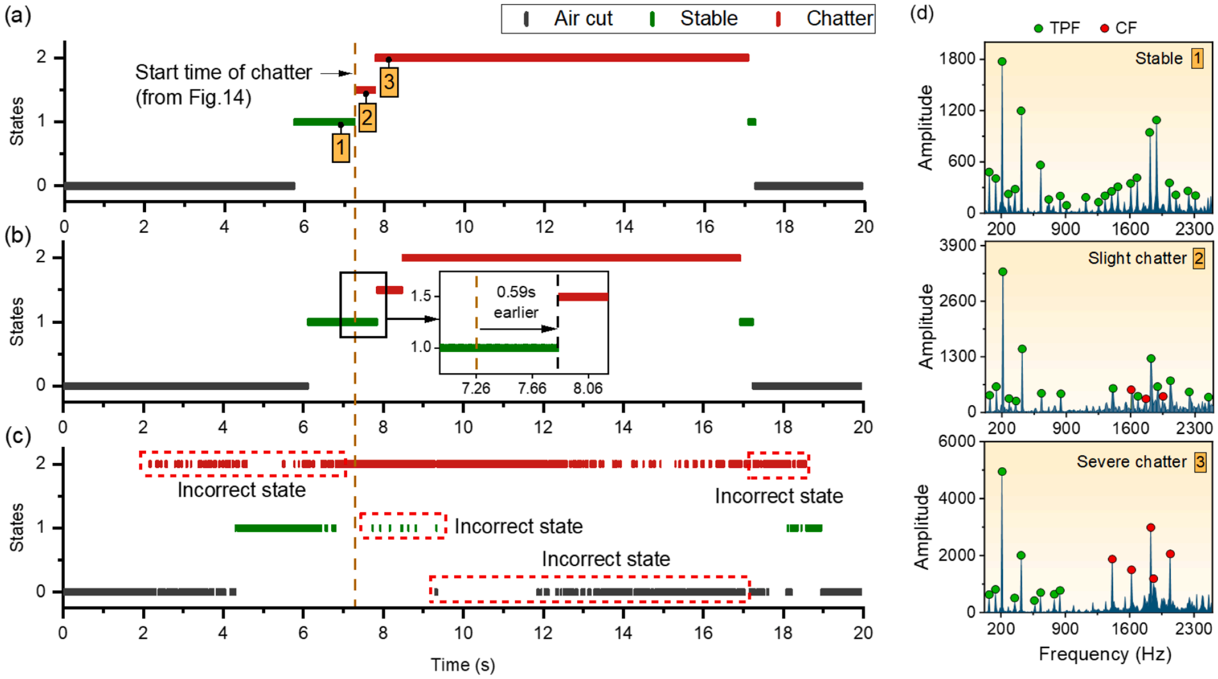

Fig. 16. States determination for No. 4: (a) ISR-CNN; (b) ResNet; (c) WPD-SVM; (d) FFT validation.

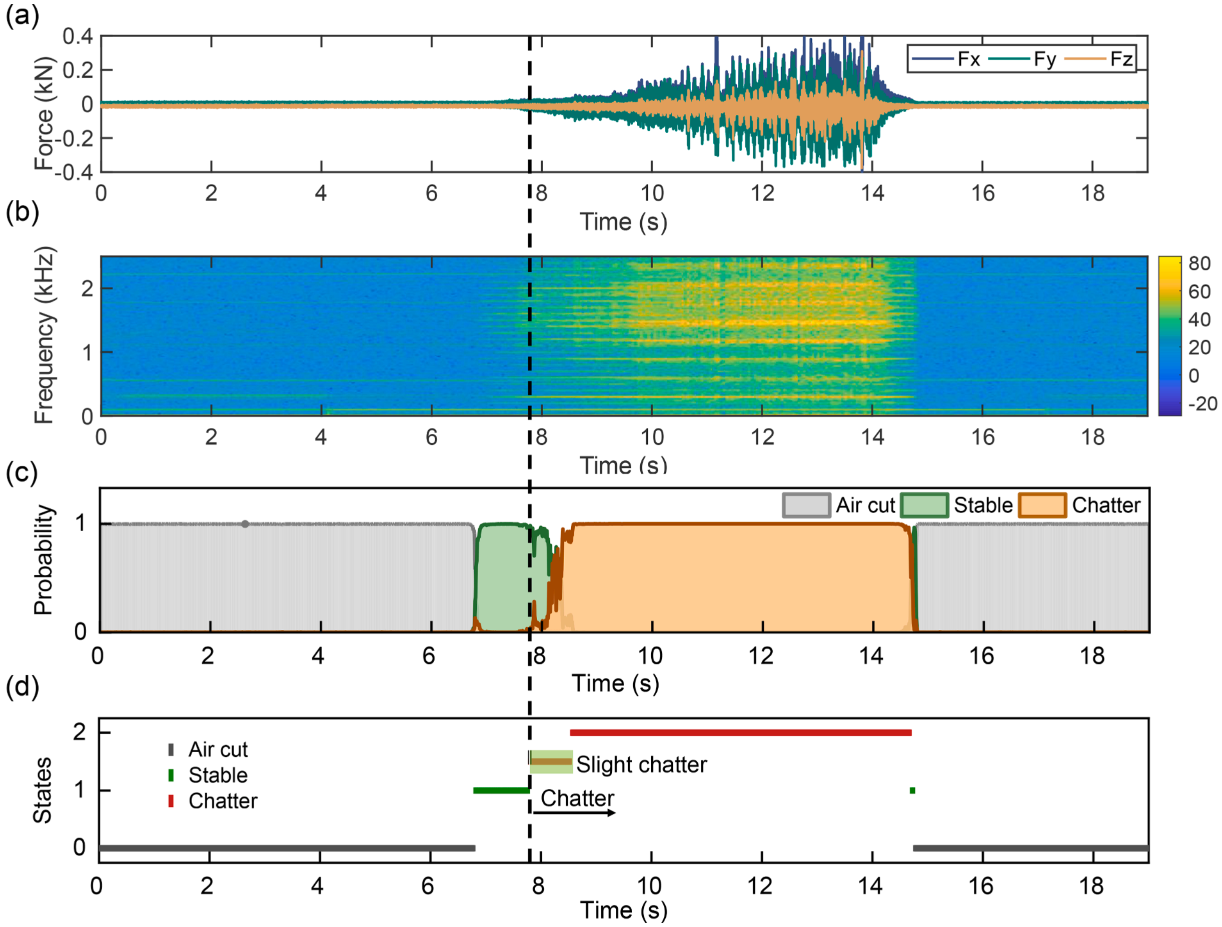

Fig. 17. Identification results of machining states for No. 5: (a) cutting force collection; (b) STFT diagram; (c) out probability; (d) states determination.

frequency of signal is the tooth passage frequency (TPF) and its multiplier frequency. When there is a slight chatter, the TPF is still dominant but the chatter frequency (CF) is already present with a small magnitude. When severe chatter occurs, the CF dominates with a large amplitude. In summary, the proposed ISR-CNN not only correctly identifies the individual machining states, but also detects the onset of chatter earlier than ResNet and WPD-SVM, which demonstrates the high discrimination accuracy of the proposed method.

4.4. *Verification of model generalization*

While the proposed method is able to accurately identify the individual machining states for No. 2, it remains to be validated for No. 4 and No. 5. This is due to the fact that No. 4 and No. 5 are not involved in model training and are only used to verify the generalisation of the model. The following is an example of the verification for No. 4, which is shown in Fig. 14. Similar to Fig. 11, the time–frequency diagram is still used. The surface quality of the workpiece is used to determine the time of onset and end of chatter as 7.3 s and 17.1 s respectively.

The obtained probabilities of each state for No. 4 relying on ISR-CNN, ResNet and WPD-SVM are presented in Fig. 15. From Fig. 15 (a) and (b), the probabilities of the ISR-CNN and ResNet outputs are close to 100 % when the machining states are in the air cut, stable and chatter states. Moreover, there are fluctuations in the probabilities at the three transition states, but the magnitude is small, which does not affect the results of the machining state identification. Conversely, Fig. 15(c) shows wide fluctuations in the probability of air cut, stable and chatter states outputs throughout the process, which would seriously affect the machining state discrimination. Therefore, it can be concluded that the chatter detection obtained by the WPD-SVM has a relatively poorer generalisation capability than the proposed ISR-CNN and ResNet.

Based on the small probability assumption principle, the machining states of No. 4 were decided as shown in Fig. 16. From Fig. 16 (a) and (b), the proposed ISR-CNN and ResNet can detect each machining state very well. The former detects the onset of chatter at 7.26 s, which is essentially the same as the moment determined in Fig. 14. The latter identifies the onset of chatter at 7.85 s, which is 0.59 s later than the proposed method. Therefore, the proposed method can identify the onset of chatter earlier, which allows more time for chatter suppression and thus avoids damage to the workpiece. From Fig. 16(c), it is observed that the detection results are

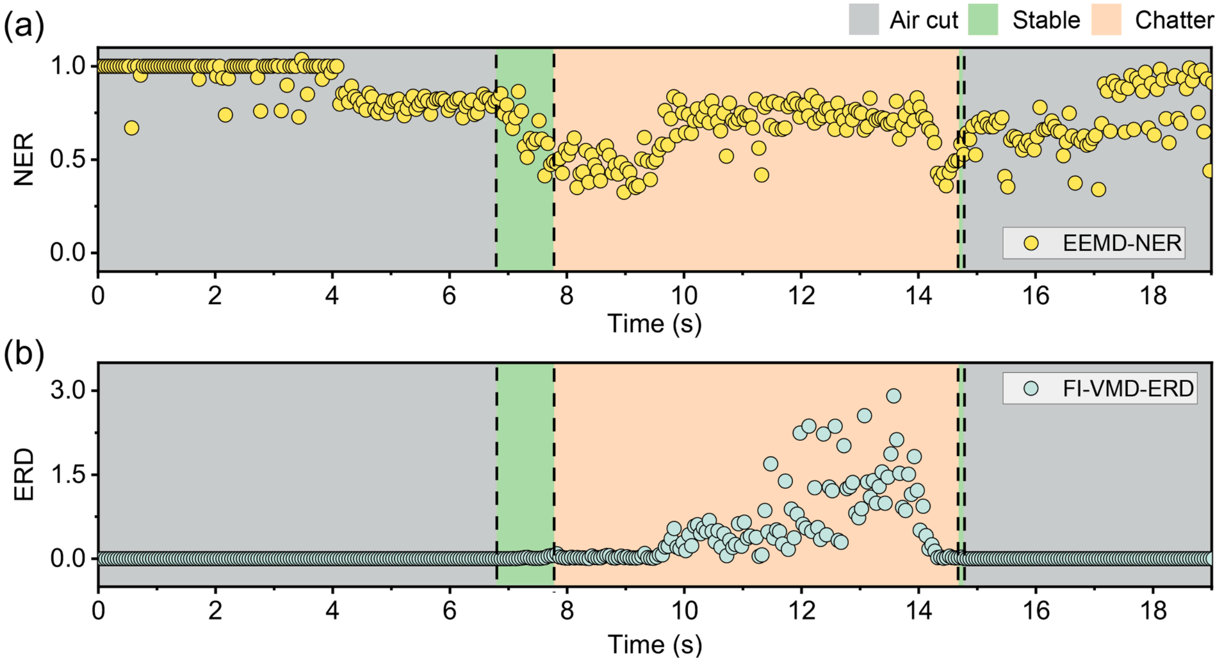

Fig. 18. Identification results of machining states for No. 5: (a) EEMD-NER; (b) FI-VMD-ERD. (Division of machining states derived from Fig. 17).

misidentified when the machining state is in the air cut, stable and chatter states. It is inferred that the established WPD-SVM model is unable to identify the machining state of No. 4, which leads to the limitations of the model. Additionally, Fig. 16(d) shows similar results to Fig. 13(d) in terms of the distinction between stable, slight chatter and severe chatter, which justifies these states being classified. In conclusion, the proposed ISR-CNN can correctly identify the machining state of No. 4, and detects the onset of chatter earlier than ResNet, which demonstrates the excellent generalization capability of the proposed method.

4.5. Comparison with other threshold methods

Over the past decade, threshold-based chatter detection methods have been emerging. As methods for time–frequency adaptive decomposition, EEMD and VMD are typical representatives and are commonly used for the decomposition of non-smooth signals. Therefore, they are often used in the field of chatter detection. Also, the energy ratio is a frequently extracted feature for the chatter detection. Therefore, EEMD and VMD combined with energy ratio are more typical methods, as has been shown in the introduction section. So literature [33] and [37] are chosen to compare with the proposed method using No. 5 as an example. Here, the former uses EEMD combined with normalised energy ratio (NER) to detect chatter. Meanwhile, the latter uses fast iterative VMD (FI-VMD) to decompose the signal and then extract the residual energy ratio (ERD) of the sub-signals. The identification results of machining states obtained with the proposed ISR-CNN for No. 5 are shown in Fig. 17. It can be observed from Fig. 17 that the proposed method can identify the air cut, stable and chatter states very well, which is intuitively consistent with the identification results of the time–frequency diagram. Moreover, based on the principle of small probability assumption, we can not only determine the time of chatter onset, but also classify the two phases of slight and severe chatter. Based on the EEMD-NER and FI-VMD-ERD methods, the identification results of machining states for No. 5 are described in Fig. 18. From Fig. 18(a) it can be observed that the distribution of NER is more scattered, which may cause the threshold value to be difficult to determine. Worse still, the NERs for the air cut and chatter states are at the same level, which would make it difficult to distinguish between these two states. In contrast, in Fig. 18(b), the ERDs of the air cut and stable states are at the same level, which makes it impossible to delineate a threshold value to distinguish between the two states. In conclusion, the proposed ISR-CNN in this paper can better identify each machining state of No. 5 while the thresholding method involved in literature [33] and [37] has difficulty in distinguishing between these machining states.

**.

5. Conclusions

Most conventional chatter detection methods require setting thresholds for chatter indicators, which is difficult to determine. Even though the extracted chatter indicators can be combined with machine learning algorithms for classification, the accuracy and generalisation of this behavioural recognition of chatter needs to be improved. Deep learning with its outstanding feature self-learning and classification capabilities can solve these problems. However, chatter detection based on deep learning relies on data labelling throughout the process, which is quite difficult for transition states. Therefore, this paper first discards the transition state data and

Table 6
Model architecture and parameters of ResNet.

Model architecture	Model Parameter	Output shape
Conv_1	7 × 1, 16, 2 × 1	(16, 125, 3)
Conv_2	7 × 1, 32, 2 × 1	(32, 63, 3)
Conv_3	7 × 1, 64, 2 × 1	(64, 32, 3)
Max pooling	2 × 1, 2 × 1	(64, 16, 3)
Residual block_1	$\begin{bmatrix}1\times1,64\\3\times3,64\\1\times1,256\end{bmatrix}$	(256, 16, 3)
Residual block_2	$\begin{bmatrix}1\times1,128\\3\times3,128\\1\times1,512\end{bmatrix}$	(512, 8, 2)
Average pooling	–	(512, 1, 1)
FC layers	512–32-16–3	(3)

*(Parameter_1, Parameter_2, Parameter_3)→(Kernel size, Channel number, Stride).

uses easily labelled data to construct the dataset. A novel ISR-CNN model is then constructed, trained and validated based on the Inception module and SR-block. Finally, the developed model is validated and tested throughout the milling experiments in order to identify machining states including transition states. The following conclusions can be drawn.

(1) The classification accuracy of the proposed ISR-CNN is 100 % and 97.9 % on the validation and test sets, respectively, which is higher than the classification accuracies obtained based on ResNet and WPD-SVM. In addition, the FI score of each machining state obtained by the proposed ISR-CNN is also the highest in both the validation and test sets, which also demonstrates the high classification accuracy of the proposed method.

(2) The clusters of individual states obtained by the proposed ISR-CNN on the validation and test sets do not overlap. Moreover, the distance between the individual clusters is further than that of ResNet and WPD-SVM, reflecting the ease of the proposed method when distinguishing between the individual states.

(3) The proposed ISR-CNN can not only correctly identify the individual machining states of milling experiments, but also detect the onset of chatter earlier than ResNet and WPD-SVM, which demonstrates the high discrimination accuracy and excellent generalization capability of the proposed method.

(4) Comparing the two thresholding methods for chatter detection (EEMD-NER and FI-SVM-ERD), the proposed ISR-CNN can better identify each machining state of milling experiment while both thresholding methods have difficulty in delineating a threshold to distinguish each machining state.

(5) The establishment of an on-line chatter detection system will go a long way towards improving productivity, ensuring machining quality and suppressing chatter. Our future work focuses on the implementation of the proposed ISR-CNN method. This refers to building an on-line chatter detection system in a factory based on the proposed method, which involves the arrangement of sensors, the integration of system and economic costs. Additional work is that we will consider a mechanistic model of chatter combined with a data-driven model to improve detection accuracy and robustness.

CRediT authorship contribution statement

Pengfei Zhang: Conceptualization, Data curation, Methodology, Software, Writing – original draft. **Dong Gao:** Resources. **Dongbo Hong:** Visualization. **Yong Lu:** Funding acquisition, Project administration, Supervision, Writing – review & editing. **Qian Wu:** Software. **Shusong Zan:** Writing – review & editing. **Zhirong Liao:** Supervision, Validation, Writing – review & editing.

Declaration of Competing Interest

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

Data availability

The authors do not have permission to share data.

Acknowledgments

This work was supported by the National Key Research and Development Project of China (2019YFB1704803).

Appendix A. .

Table 6. Model architecture and parameters of ResNet.

Appendix B. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.ymssp.2023.110241.

References

[1] L. Zhu, C. Liu, Recent progress of chatter prediction, detection and suppression in milling, Mech. Syst. Signal Process. 143 (2020), 106840, https://doi.org/10.1016/j.ymssp.2020.106840.

[2] G. Quintana, J. Ciurana, Chatter in machining processes: A review, Int. J. Mach. Tools Manuf. 51 (2011) 363–376, https://doi.org/10.1016/j.ijmachtools.2011.01.001.

[3] D. Axinte, H. Huang, J. Yan, Z. Liao, What micro-mechanical testing can reveal about machining processes, Int. J. Mach. Tools Manuf. 183 (2022), 103964, https://doi.org/10.1016/j.ijmachtools.2022.103964.

[4] G. Totis, M. Sortino, Polynomial Chaos-Kriging approaches for an efficient probabilistic chatter prediction in milling, Int. J. Mach. Tools Manuf. 157 (2020), 103610, https://doi.org/10.1016/j.ijmachtools.2020.103610.

[5] M. Cordes, W. Hintze, Y. Altintas, Chatter stability in robotic milling, Robot. Comput. Integr. Manuf. 55 (2019) 11–18, https://doi.org/10.1016/j.rcim.2018.07.004.

[6] M. Postel, O. Özsahin, Y. Altintas, High speed tooltip FRF predictions of arbitrary tool-holder combinations based on operational spindle identification, Int. J. Mach. Tools Manuf. 129 (2018) 48–60, https://doi.org/10.1016/j.ijmachtools.2018.03.004.

[7] J. Munoa, X. Beudaert, Z. Dombovari, Y. Altintas, E. Budak, C. Brecher, G. Stepan, Chatter suppression techniques in metal cutting, CIRP Ann. - Manuf. Technol. 65 (2016) 785–808, https://doi.org/10.1016/j.cirp.2016.06.004.

[8] X. Beudaert, K. Erkorkmaz, J. Munoa, Portable damping system for chatter suppression on flexible workpieces, CIRP Ann. 68 (2019) 423–426, https://doi.org/10.1016/j.cirp.2019.04.010.

[9] X. Zhang, C. Wang, J. Liu, R. Yan, H. Cao, X. Chen, Robust active control based milling chatter suppression with perturbation model via piezoelectric stack actuators, Mech. Syst. Signal Process. 120 (2019) 808–835, https://doi.org/10.1016/j.ymssp.2018.10.043.

[10] M. Ntemi, S. Paraschos, A. Karakostas, I. Gialampoukidis, S. Vrochidis, I. Kompatsiaris, Infrastructure monitoring and quality diagnosis in CNC machining: A review, CIRP J. Manuf. Sci. Technol. 38 (2022) 631–649, https://doi.org/10.1016/j.cirpj.2022.06.001.

[11] W.K. Wang, M. Wan, W.H. Zhang, Y. Yang, Chatter detection methods in the machining processes: A review, J. Manuf. Process. 77 (2022) 240–259, https://doi.org/10.1016/j.jmapro.2022.03.018.

[12] Z. Zhang, H. Li, G. Meng, X. Tu, C. Cheng, Chatter detection in milling process based on the energy entropy of VMD and WPD, Int. J. Mach. Tools Manuf. 108 (2016) 106–112, https://doi.org/10.1016/j.ijmachtools.2016.06.002.

[13] Y. Dun, L. Zhus, B. Yan, S. Wang, A chatter detection method in milling of thin-walled TC4 alloy workpiece based on auto-encoding and hybrid clustering, Mech. Syst. Signal Process. 158 (2021) 1–23, https://doi.org/10.1016/j.ymssp.2021.107755.

[14] R. Wang, Q. Song, Z. Liu, H. Ma, Z. Liu, Multi-condition identification in milling Ti-6Al-4V thin-walled parts based on sensor fusion, Mech. Syst. Signal Process. 164 (2022), 108264, https://doi.org/10.1016/j.ymssp.2021.108264.

[15] B. Liu, C. Liu, Y. Zhou, D. Wang, Y. Dun, An unsupervised chatter detection method based on AE and merging GMM and K-means, Mech. Syst. Signal Process. 186 (2023), 109861, https://doi.org/10.1016/j.ymssp.2022.109861.

[16] J. Tao, C. Qin, D. Xiao, H. Shi, X. Ling, B. Li, C. Liu, Timely chatter identification for robotic drilling using a local maximum synchrosqueezing-based method, J. Intell. Manuf. 31 (2020) 1243–1255, https://doi.org/10.1007/s10845-019-01509-5.

[17] Y. Chen, H. Li, L. Hou, J. Wang, X. Bu, An intelligent chatter detection method based on EEMD and feature selection with multi-channel vibration signals, Meas. J. Int. Meas. Confed. 127 (2018) 356–365, https://doi.org/10.1016/j.measurement.2018.06.006.

[18] D. Chen, X. Zhang, H. Zhao, H. Ding, Development of a novel online chatter monitoring system for flexible milling process, Mech. Syst. Signal Process. 159 (2021), 107799, https://doi.org/10.1016/j.ymssp.2021.107799.

[19] Y. Ren, Y. Ding, Online milling chatter identification using adaptive Hankel low-rank decomposition, Mech. Syst. Signal Process. 169 (2022), 108758, https://doi.org/10.1016/j.ymssp.2021.108758.

[20] X. Wang, Q. Song, Z. Liu, Precise chatter monitoring of thin-walled component milling process based on parametric time-frequency transform method, J. Mater. Process. Technol. 283 (2020), 116712, https://doi.org/10.1016/j.jmatprotec.2020.116712.

[21] Y. Liu, X. Wang, J. Lin, X. Kong, An adaptive grinding chatter detection method considering the chatter frequency shift characteristic, Mech. Syst. Signal Process. 142 (2020), 106672, https://doi.org/10.1016/j.ymssp.2020.106672.

[22] D. Aslan, Y. Altintas, On-line chatter detection in milling using drive motor current commands extracted from CNC, Int. J. Mach. Tools Manuf. 132 (2018) 64–80, https://doi.org/10.1016/j.ijmachtools.2018.04.007.

[23] A. Devillez, D. Dudzinski, Tool vibration detection with eddy current sensors in machining process and computation of stability lobes using fuzzy classifiers, Mech. Syst. Signal Process. 21 (2007) 441–456, https://doi.org/10.1016/j.ymssp.2005.11.007.

[24] M. Lamraoui, M. Thomas, M. El Badaoui, F. Girardin, Indicators for monitoring chatter in milling based on instantaneous angular speeds, Mech. Syst. Signal Process. 44 (2014) 72–85, https://doi.org/10.1016/j.ymssp.2013.05.002.

[25] Z. Han, Y. Zhuo, Y. Yan, H. Jin, H. Fu, Chatter detection in milling of thin-walled parts using multi-channel feature fusion and temporal attention-based network, Mech. Syst. Signal Process. 179 (2022), 109367, https://doi.org/10.1016/j.ymssp.2022.109367.

[26] S. Yan, Y. Sun, Early chatter detection in thin-walled workpiece milling process based on multi-synchrosqueezing transform and feature selection, Mech. Syst. Signal Process. 169 (2022), 108622, https://doi.org/10.1016/j.ymssp.2021.108622.

[27] S. Tangjitsitcharoen, T. Saksri, S. Ratanakuakangwan, Advance in chatter detection in ball end milling process by utilizing wavelet transform, J. Intell. Manuf. 26 (2015) 485–499, https://doi.org/10.1007/s10845-013-0805-3.

[28] B. Yang, K. Guo, Q. Zhou, J. Sun, Early chatter detection in robotic milling under variable robot postures and cutting parameters, Mech. Syst. Signal Process. 186 (2023), 109860, https://doi.org/10.1016/j.ymssp.2022.109860.

[29] Y. Sun, Z. Xiong, An optimal weighted wavelet packet entropy method with application to real-time chatter detection, IEEE/ASME Trans. Mechatronics. 21 (2016) 2004–2014, https://doi.org/10.1109/TMECH.2016.2547481.

[30] Y. Ji, X. Wang, Z. Liu, H. Wang, L. Jiao, D. Wang, S. Leng, Early milling chatter identification by improved empirical mode decomposition and multi-indicator synthetic evaluation, J. Sound Vib. 433 (2018) 138–159, https://doi.org/10.1016/j.jsv.2018.07.019.

[31] X. Zheng, P. Arrazola, R. Perez, D. Echebarria, D. Kiritsis, P. Aristimuño, M. Sáez-de-Buruaga, Exploring the effectiveness of using internal CNC system signals for chatter detection in milling process, Mech. Syst. Signal Process. 185 (2023), 109812, https://doi.org/10.1016/j.ymssp.2022.109812.

[32] H. Cao, Y. Lei, Z. He, Chatter identification in end milling process using wavelet packets and Hilbert-Huang transform, Int. J. Mach. Tools Manuf. 69 (2013) 11–19, https://doi.org/10.1016/j.ijmachtools.2013.02.007.

[33] Y. Fu, Y. Zhang, H. Zhou, D. Li, H. Liu, H. Qiao, X. Wang, Timely online chatter detection in end milling process, Mech. Syst. Signal Process. 75 (2016) 668–688, https://doi.org/10.1016/j.ymssp.2016.01.003.

[34] C. Liu, L. Zhu, C. Ni, Chatter detection in milling process based on VMD and energy entropy, Mech. Syst. Signal Process. 105 (2018) 169–182, https://doi.org/10.1016/j.ymssp.2017.11.046.

[35] K. Yang, G. Wang, Y. Dong, Q. Zhang, L. Sang, Early chatter identification based on an optimized variational mode decomposition, Mech. Syst. Signal Process. 115 (2019) 238–254, https://doi.org/10.1016/j.ymssp.2018.05.052.

[36] H. Cao, K. Zhou, X. Chen, Chatter identification in end milling process based on EEMD and nonlinear dimensionless indicators, Int. J. Mach. Tools Manuf. 92 (2015) 52–59, https://doi.org/10.1016/j.ijmachtools.2015.03.002.

[37] P. Zhang, D. Gao, Y. Lu, L. Kong, Z. Ma, Online chatter detection in milling process based on fast iterative VMD and energy ratio difference, Meas. J. Int. Meas. Confed. 194 (2022), 111060, https://doi.org/10.1016/j.measurement.2022.111060.

[38] Y. Hao, L. Zhu, B. Yan, S. Qin, D. Cui, H. Lu, Milling chatter detection with WPD and power entropy for Ti-6Al-4V thin-walled parts based on multi-source signals fusion, Mech. Syst. Signal Process. 177 (2022), 109225, https://doi.org/10.1016/j.ymssp.2022.109225.

[39] Y. Sun, L. Ding, C. Liu, Z. Xiong, X. Zhu, Beat Effect in Machining Chatter: Analysis and Detection, J. Manuf. Sci. Eng. Trans. ASME. 143 (2021), https://doi.org/10.1115/1.4047736/1085074.

[40] H. Cao, K. Zhou, X. Chen, X. Zhang, Early chatter detection in end milling based on multi-feature fusion and 3σ criterion, Int. J. Adv. Manuf. Technol. 92 (2017) 4387–4397, https://doi.org/10.1007/s00170-017-0476-x.

[41] Z. Yao, D. Mei, Z. Chen, On-line chatter detection and identification based on wavelet and support vector machine, J. Mater. Process. Technol. 210 (2010) 713–719, https://doi.org/10.1016/j.jmatprotec.2009.11.007.

[42] I. Oleaga, C. Pardo, J.J. Zulaika, A. Bustillo, A machine-learning based solution for chatter prediction in heavy-duty milling machines, Meas. J. Int. Meas. Confed. 128 (2018) 34–44, https://doi.org/10.1016/j.measurement.2018.06.028.

[43] K. Li, S. He, B. Li, H. Liu, X. Mao, C. Shi, A novel online chatter detection method in milling process based on multiscale entropy and gradient tree boosting, Mech. Syst. Signal Process. 135 (2020), 106385, https://doi.org/10.1016/j.ymssp.2019.106385.

[44] Y. Wang, M. Zhang, X. Tang, F. Peng, R. Yan, A kMap optimized VMD-SVM model for milling chatter detection with an industrial robot, J. Intell. Manuf. 33 (2022) 1483–1502, https://doi.org/10.1007/s10845-021-01736-9.

[45] M.Q. Tran, M.K. Liu, M. Elsisi, Effective multi-sensor data fusion for chatter detection in milling process, ISA Trans. 125 (2022) 514–527, https://doi.org/10.1016/j.isatra.2021.07.005.

[46] M.C. Yesilli, F.A. Khasawneh, A. Otto, On transfer learning for chatter detection in turning using wavelet packet transform and ensemble empirical mode decomposition, CIRP J. Manuf. Sci. Technol. 28 (2020) 118–135, https://doi.org/10.1016/j.cirpj.2019.11.003.

[47] R. Zhao, R. Yan, Z. Chen, K. Mao, P. Wang, R.X. Gao, Deep learning and its applications to machine health monitoring, Mech. Syst. Signal Process. 115 (2019) 213–237, https://doi.org/10.1016/j.ymssp.2018.05.050.

[48] Y. Lei, N. Li, L. Guo, N. Li, T. Yan, J. Lin, Machinery health prognostics: A systematic review from data acquisition to RUL prediction, Mech. Syst. Signal Process. 104 (2018) 799–834, https://doi.org/10.1016/j.ymssp.2017.11.016.

[49] Y. Lei, B. Yang, X. Jiang, F. Jia, N. Li, A.K. Nandi, Applications of machine learning to machine fault diagnosis: A review and roadmap, Mech. Syst. Signal Process. 138 (2020), 106587, https://doi.org/10.1016/j.ymssp.2019.106587.

[50] P. Zhang, D. Gao, Y. Lu, Z. Ma, X. Wang, X. Song, Cutting tool wear monitoring based on a smart toolholder with embedded force and vibration sensors and an improved residual network, Meas. J. Int. Meas. Confed. 199 (2022), 111520, https://doi.org/10.1016/j.measurement.2022.111520.

[51] Y. Sun, J. He, H. Ma, X. Yang, Z. Xiong, X. Zhu, Y. Wang, Online chatter detection considering beat effect based on Inception and LSTM neural networks, Mech. Syst. Signal Process. 184 (2023), 109723, https://doi.org/10.1016/j.ymssp.2022.109723.

[52] R.K. Vashisht, Q. Peng, Online chatter detection for milling operations using LSTM neural networks assisted by motor current signals of ball screw drives, J. Manuf. Sci. Eng. Trans. ASME. 143 (2021), https://doi.org/10.1115/1.4048001.

[53] B. Sener, M.U. Gudelek, A.M. Ozbayoglu, H.O. Unver, A novel chatter detection method for milling using deep convolution neural networks, Meas. J. Int. Meas. Confed. 182 (2021) 109689.

[54] M.H. Rahimi, H.N. Huynh, Y. Altintas, On-line chatter detection in milling with hybrid machine learning and physics-based model, CIRP J. Manuf. Sci. Technol. 35 (2021) 25–40, https://doi.org/10.1016/j.cirpj.2021.05.006.

[55] S. Wan, X. Li, Y. Yin, J. Hong, Milling chatter detection by multi-feature fusion and Adaboost-SVM, Mech. Syst. Signal Process. 156 (2021), 107671, https://doi.org/10.1016/j.ymssp.2021.107671.

[56] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proc. IEEE. 86 (1998) 2278–2323, https://doi.org/10.1109/5.726791.

[57] D. Zhou, X. Zhuang, H. Zuo, J. Cai, X. Zhao, J. Xiang, A model fusion strategy for identifying aircraft risk using CNN and Att-BiLSTM, Reliab. Eng. Syst. Saf. 228 (2022), 108750, https://doi.org/10.1016/j.ress.2022.108750.

[58] A. Krizhevsky, I. Sutskever, G.E. Hinton, ImageNet classification with deep convolutional neural networks, Commun. ACM. 60 (2017) 84–90, https://doi.org/10.1145/3065386.

[59] K. Simonyan, A. Zisserman, Very Deep Convolutional Networks for Large-Scale Image Recognition, 3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc. (2014). https://doi.org/10.48550/arxiv.1409.1556.

[60] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going Deeper with Convolutions, IEEE Conf. Comput. Vis. Patt. Recogn. (2015) 1–9.

[61] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, Rethinking the Inception Architecture for Computer Vision, Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (2016) 2818–2826. https://doi.org/10.1109/CVPR.2016.308.

[62] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (2016) 770–778. https://doi.org/10.1109/CVPR.2016.90.

[63] H. Jégou, M. Douze, C. Schmid, Product quantization for nearest neighbor search, IEEE Trans. Pattern Anal. Mach. Intell. 33 (2011) 117–128, https://doi.org/10.1109/TPAMI.2010.57.

[64] J. Hu, L. Shen, G. Sun, Squeeze-and-Excitation Networks, Proc. IEEE Conf. Comput. Vis, Pattern Recognit. (2018) 7132–7141.

[65] B. Yan, L. Zhu, Y. Dun, Tool wear monitoring of TC4 titanium alloy milling process based on multi-channel signal and time-dependent properties by using deep learning, J. Manuf. Syst. 61 (2021) 495–508, https://doi.org/10.1016/j.jmsy.2021.09.017.

[66] Z. Liu, B. Guo, F. Wu, T. Han, L. Zhang, An Improved Burr Size Prediction Method Based on the 1d-Resnet Model and Transfer Learning, J. Manuf. Process. 84 (2022) 183–197, https://doi.org/10.2139/ssrn.4092491.

[67] M.C. Yesilli, F.A. Khasawneh, B.P. Mann, Transfer learning for autonomous chatter detection in machining, J. Manuf. Process. 80 (2022) 1–27, https://doi.org/10.1016/J.JMAPRO.2022.05.037.

[68] Y. Fu, Y. Zhang, H. Gao, T. Mao, H. Zhou, R. Sun, D. Li, Automatic feature constructing from vibration signals for machining state monitoring, J. Intell. Manuf. 30 (2019) 995–1008, https://doi.org/10.1007/s10845-017-1302-x.