sensors

Article

An Energy Data-Driven Approach for Operating Status Recognition of Machine Tools Based on Deep Learning

Wei Yan [1,2], Chenxun Lu [3,4], Ying Liu [2,*], Xumei Zhang [1] and Hua Zhang [4]

[1] School of Automobile and Traffic Engineering, Wuhan University of Science and Technology, Wuhan 430081, China
[2] Department of Mechanical Engineering, School of Engineering, Cardiff University, Cardiff CF24 3AA, UK
[3] Hubei Key Laboratory of Mechanical Transmission and Manufacturing Engineering, Wuhan University of Science and Technology, Wuhan 430081, China
[4] Academy of Green Manufacturing Engineering, Wuhan University of Science and Technology, Wuhan 430081, China
* Correspondence: liuy81@cardiff.ac.uk

Abstract: Machine tools, as an indispensable equipment in the manufacturing industry, are widely used in industrial production. The harsh and complex working environment can easily cause the failure of machine tools during operation, and there is an urgent requirement to improve the fault diagnosis ability of machine tools. Through the identification of the operating state (OS) of the machine tools, defining the time point of machine tool failure and the working energy-consuming unit can be assessed. In this way, the fault diagnosis time of the machine tool is shortened and the fault diagnosis ability is improved. Aiming at the problems of low recognition accuracy, slow convergence speed and weak generalization ability of traditional OS recognition methods, a deep learning method based on data-driven machine tool OS recognition is proposed. Various power data (such as signals or images) of CNC machine tools can be used to recognize the OS of the machine tool, followed by an intuitive judgement regarding whether the energy-consuming units included in the OS are faulty. First, the power data are collected, and the data are preprocessed by noise reduction and cropping using the data preprocessing method of wavelet transform (WT). Then, an AlexNet Convolutional Neural Network (ACNN) is built to identify the OS of the machine tool. In addition, a parameter adaptive adjustment mechanism of the ACNN is studied to improve identification performance. Finally, a case study is presented to verify the effectiveness of the proposed approach. To illustrate the superiority of this method, the approach was compared with traditional classification methods, and the results reveal the superiority in the recognition accuracy and computing speed of this AI technology. Moreover, the technique uses power data as a dataset, and also demonstrates good progress in portability and anti-interference.

Keywords: operating status recognition; energy data-driven; deep learning; machine tools; fault diagnosis

Citation: Yan, W.; Lu, C.; Liu, Y.; Zhang, X.; Zhang, H. An Energy Data-Driven Approach for Operating Status Recognition of Machine Tools Based on Deep Learning. *Sensors* **2022**, *22*, 6628. https://doi.org/10.3390/s22176628

Academic Editors: Long Wen, Haidong Shao, Xinyu Li and Zhuyun Chen

Received: 7 August 2022
Accepted: 29 August 2022
Published: 1 September 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

1. Introduction

Intelligent computer numerically controlled (CNC) machine tools are necessary pieces of equipment for the furthering the development of modern productivity [1]. According to incomplete statistics, the number of machine tools in China has exceeded 8 million [2]. A CNC machine tool is a complex system integrating mechanical, electrical, hydraulic and other technologies. At the same time, some studies have shown that the complexity of the system structure leads to the cascading of its failures. Even if the failure rate of one component is low, the overall failure rate of the system increases exponentially due to the scale effect of complex systems. This renders maintenance of machine tools increasingly difficult. The fault diagnosis of machine tools in operation plays an irreplaceable role in ensuring the running quality of machine tools, especially in maintaining the level of machining accuracy [3]. Therefore, improving the fault diagnosis capability will significantly improve the life cycle of the machine tools, and has become a central focus within relevant industries and academia. When a machine tool fails, it is necessary to identify the time of failure and the failed energy-consuming unit as soon as possible. Fu et al. [4] found that this time accounted for 80–90% of the total fault diagnosis time, which was not conducive to subsequent fault diagnosis.

The OS of the machine tools is influenced by their energy-consuming units. By accurately recognizing the OS of the machine tool, it is possible to accurately analyze the energy-consuming units that are working at any time. This reduces the time required for locking the time point and location of the fault, which plays an important role in improving the efficiency of system fault diagnosis and represents preparatory work for fault diagnosis. Existing research shows that the operating state (OS) of a machine tool can be divided into cutting, starting, standby, etc., [5,6]. This paper uses the power curve of the XK713 milling machine operating process, as shown in Figure 1. The energy-consuming units contained within the power of each OS can be visually observed. For example, in the cutting state, almost all the energy units of a machine tool are running, such as the spindle drive system, feed drive system, lubrication and cooling system, etc., and only three energy units are in a standby state. When the machine tool fails, by determining the OS of the machine tool, the energy-consuming unit where the failure is located can be identified. However, due to the instability of the machining site and the accuracy of the measuring instruments, it is difficult to establish a specific model to accurately recognize the OS of the machine tool. For example, the load factor is a key factor in calculating the energy consumption in the cutting state, which needs to be obtained by fitting the experimental data of additional sensors, which is difficult to apply in actual power production [7]. To this end, a deep learning method with power data is proposed to recognize the OS of machine tools in this paper, which solves the problems of large amounts of machine tool data and a complex working environment, and demonstrates good recognition accuracy and generalization ability.

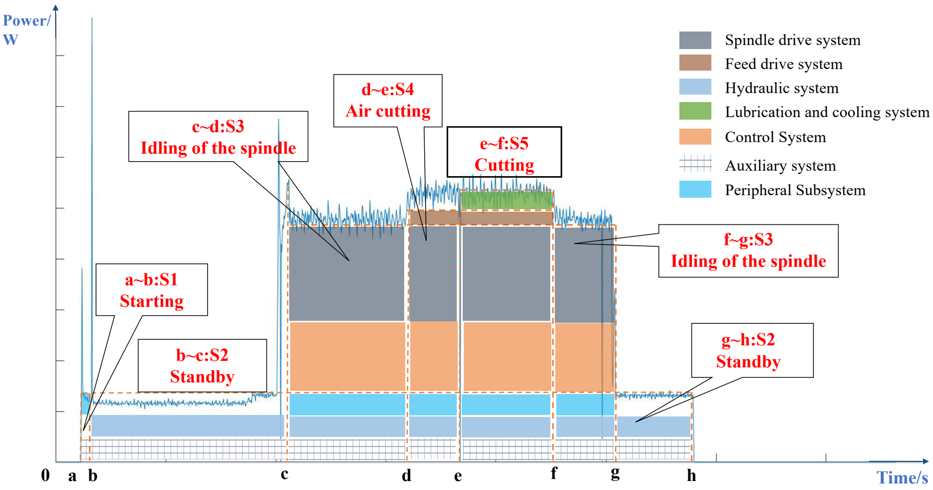


Figure 1. Power curve of the XK713 machine tool process.

The rest of the paper is structured as follows. Section 2 describes the related work for the OS recognition of a machine tool. Section 3 introduces the methodology of the proposed approach, based on the general framework establishment, whereby the three main steps, namely power data acquisition, OS recognition with deep learning technology ACNN and model validation are also studied. In Section 4, a case study is applied to verify the feasibility and superiority of the proposed method. Then, conclusions and further research directions are outlined in Section 5.

2. Literature Review

An essential basis for this study is to classify and recognize the OS of a machine tool. In general, the OS is a term to describe the series of activities of machining tools [8,9], and can be divided into five stages, namely Starting status (S1), Standby status (S2), Idling of the spindle status (S3), Air cutting status (S4) and Cutting status (S5) [10]. With a defined boundary of OS, several researchers focus on the energy-consumption nature of machine tools and their energy-consuming units. Oliver et al. divided the OS of a machine tool into the steady state and transient state, and estimated the energy consumption of the spindle and feed axis with these states [11,12]. Lv et al. established a spindle acceleration state model of CNC lathes based on a calculation of the moment of inertia for the spindle drive system [13]. Martin et al. conducted an experimental study to investigate the energy characteristics required for obtaining the power models of CNC machine tools through an experimental study [14]. These studies provide the fundamental theory for this work.

When the boundaries and energy-consumption characteristics of OS are determined, the power thresholds can be set to recognize the OS of a machine tool. There are two main approaches for power threshold acquisition in the existing studies. One is to measure the key parameters of the OS energy consumption, and the other concerns calculation of the power thresholds in each OS by using the parameters [15]. Hu et al. designed a spindle speed experiment to obtain the non-load power factor, which was used to calculate the power thresholds of S4 [16]. As a key parameter for S5, the load-loss coefficients during the machine tool service process (MTSP) were calculated by using a data-fitting approach, and the data were collected from the torque sensor, power sensor and power meters in the machining experiment [17]. Benjamin et al. proposed a Gaussian threshold method to adjust the judgment threshold for different machine tool OS recognition [18]. Liu et al. designed a machining experiment of the machine tool spindle drive system, and calculated the load-loss coefficients [19]. The aforementioned power-threshold approaches provide strong support for understanding the energy-consumption characteristics of OS and the complex relationship with energy-consuming units, which contribute to the study of OS recognition in this paper. Unfortunately, most of the threshold methods require installation of additional sensors in the processing system, but are difficult to widely implement in actual industrial production processes and also lack generalizability.

The other is to establish the multi-period energy-consumption model of energy units, where the power thresholds of OS could then be obtained by summing the results of these models. Due to the complex structure of machine tools, the energy-consumption models of the feed drive system [20,21], spindle drive system [22,23] and fluidic systems [24] were established. He et al. combined the colored timed object-oriented Petri net (CTOPN) and the virtual component method to establish the dynamic characteristic model of multiple energy sources of CNC machine tools [25]. Jia et al. established a kinetic energy-consumption model [26]. The above two models are based on the energy consumption of the basic actions of machine tools, and the energy consumption of the OS can be obtained by the addition of the two values of consumption. The above approaches often depend on a large number of experiments to improve the recognition accuracy, and are inconvenient for implementation and operation in actual production. Furthermore, the number of energy units is increasing due to the continuous structural improvements of machine tools, and the above approaches are often labor-intensive and computationally costly.

Traditional machine tool fault diagnosis mainly uses signal processing and machine learning techniques. The signal processing techniques used in machine tool fault diagnosis mainly include time-domain analysis [27], frequency-domain analysis [28] and time-frequency analysis. Wavelet Analysis [29], short-time Fourier transform (STFT) [30] and empirical mode decomposition are commonly used methods in the time-frequency analysis of machine tool signals. In recent years, the convolutional neural network (CNN) has been used in the field of image recognition. Great achievements have been achieved in this domain, which can be extracted from the original complex and robust multi-dimensional features that are extracted from raw data [31,32], so that the established model has higher accuracy and robustness. Wan et al. [33] proposed an improved two-dimensional LeNet-5 network for monitoring and identification of rolling bearing faults. Shi et al. [34] use the DCNN algorithm to identify the effective operating state of the machine tool. In the above methods and models, the adaptive feature extraction of original data by CNN only considers the multi-dimensional characteristics of the data, and does not consider the problems of data generalization and gradient explosion, which result in losses of original data feature information. Rohit et al. [35] use a modified AlexNet in conjunction with the Grasshopper Optimization Algorithm (IGOA) to diagnose faults present in gearboxes and to correct the faulty gear. In addition, after the classification process, performance evaluation is performed on various performance indicators. Existing deep learning [36] methods are processed and optimized to improve its performance. However, due to the constraints of experimental conditions, the evaluation cost is higher, and large deep convolutional networks such as VGGNet and ResNet-18 that use more memory and parameters (140 M) have not been adopted [37].

Therefore, this paper proposes a data-processing method based on WT, which only needs to collect power data during operation of the machine tool, and the collection method is convenient and fast. Further, the proposed method does not require the use of additional hardware measurements. It can be used as a general method to determine the current OS of various machine tools, thus ensuring its practical application in manufacturing enterprises. Finally, the paper adopts an ACNN model with more hidden layers, which employs multiple GPU operations [38] and an advanced activation function ReLU, capable of extracting deeper features from the data.

3. Materials and Methods

3.1. General Framework

The general framework of the proposed approach includes three main stages, power data collection and processing, recognition model establishment and model validation, as shown in Figure 2.

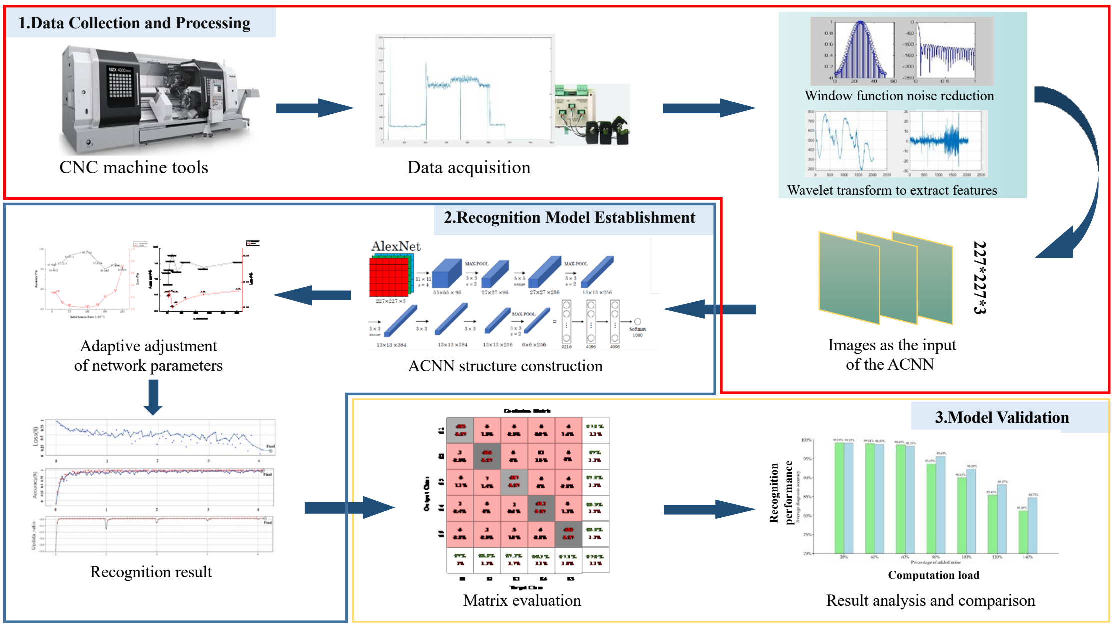

1.Data Collection and Processing
CNC machine tools
Data acquisition
Window function noise reduction
Wavelet transform to extract features
227*227*3
Images as the input of the ACNN
2.Recognition Model Establishment
AlexNet
ACNN structure construction
Adaptive adjustment of network parameters
Recognition result
3.Model Validation
Matrix evaluation
Recognition performance
Computation load
Result analysis and comparison

Figure 2. The general framework of the proposed approach.

- Power data collection and processing. The purpose of this stage is to obtain the input of ACNN, which mainly includes three parts: power data acquisition, power data preprocessing and feature extraction.
- Recognition model establishment. In this stage, an AI recognition model for OS of machine tools is established with ACNN. Then, the adjustment approach for hyperparameters of ACNN is also studied to improve the recognition performance.
- Model validation. To illustrate the feasibility of the above model, a confusion matrix approach is employed to verify the recognition accuracy.

3.2. Power Data Collection and Processing

3.2.1. Power Data Collection and Preprocessing

The dataset in this study comes from the Green Manufacturing Laboratory of Wuhan University of Science and Technology. We use the power tester to collect the power data of the entire processing process of the milling machine XK713 from start to finish, that is, the three-phase current signal during processing, and then display it on the host computer. In order to avoid the influence of a single number of processing machines on the test and to increase the stability of the acquisition environment, the entire test process was carried out on 20 machine tools of the same model. A total of 400 blank milling tests were performed.

In addition, it should be noted that there are various sources of noise present in the original current signal, which may cause serious interference to the main components of the current signal and reduce the quality of the data set. In order to improve the signal-to-noise ratio, this paper uses a rectangular window function to smooth the original data. Considering the power curve of the OS of machine tools, the sine function could be used as the time-domain prototype formula of the filter window, as shown in Equation (1).

$$y = \frac{\sin(2\pi fx)}{\pi x}, \tag{1}$$

where, f is the cut-off frequency of filter, which is used to describe a special frequency of the frequency characteristic index; x is the current signal value collected; y is the signal value after filtering and transformation.

In addition, due to the time-series characteristics of the current signal, the time-domain formula of the filter window is constructed with Equation (2).

$$X(k) \Rightarrow x(i) \cdot \frac{\sin(2\pi f(i - \frac{L-1}{2}))}{\pi(i - \frac{L-1}{2})}, \tag{2}$$

where, L represents the window function width; i represents the sample value sequence number, and $i \in [0, L-1]$; $X(k)$ is the filter sequence; $x(i)$ is the current sampling point. L and $x(i)$ depend on the step size and sampling frequency and are the key to smooth denoising of the raw data.

3.2.2. Feature Extraction

The correct input form of the ACNN is a prerequisite to ensure the reasonable use of the entire model. Time-domain characteristics can reflect the timely and dynamic processing of the machine tool and the trend and characteristics of signal changes during processing. Therefore, the continuous WT approach is employed to extract features of the processed signal to time-frequency images, and these images can be used as input for the ACNN. The continuous WT function is shown in Equation (3).

$$\psi(\alpha,\tau)=\frac{1}{\sqrt{\alpha}}\int_{-\infty}^{\infty} f(t)\cdot\psi(\frac{t-\tau}{\alpha})\cdot dt\,, \tag{3}$$

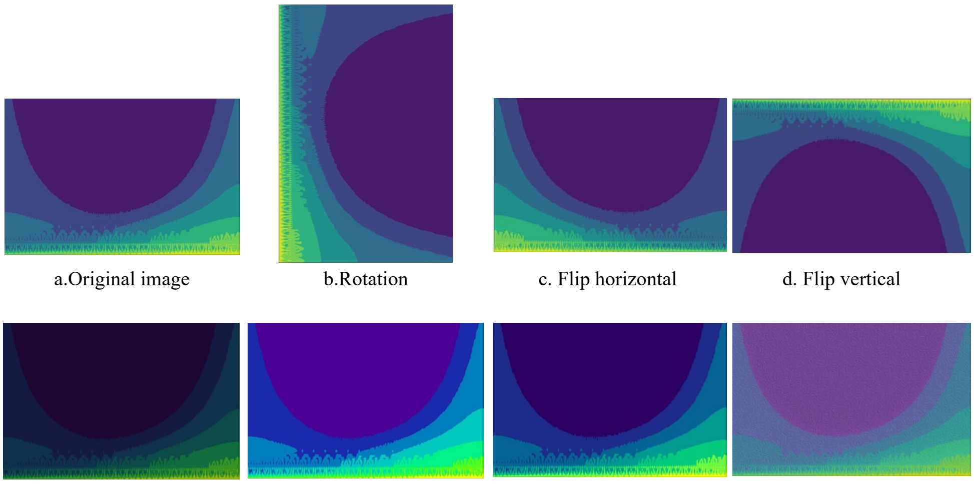
where, $\psi(\alpha,\tau)$ is the wavelet function; α is the scale factor; τ is the translation factor and $f(t)$ represents the commonly used piecewise function, and is the collected current signal in this paper.

Subsequently, the wavelet time-frequency images are stored in .jpg format, and resized to the pixel-based dimensions of 227 × 227, which can be used as input for the ACNN.

3.2.3. Data Augmentation

The time-frequency diagram of the processed current signal is intercepted. Next, 5 different categories of sample images are generated (corresponding to 5 OS, respectively), each group of 400 images, a total of 2000 images.

In order to improve the adaptability and generalization of the network model, this study uses random rotation, horizontal and vertical flipping, adjusting brightness, saturation and contrast and adding Gaussian blur to the obtained images to expand the dataset [39]. The number of augmented datasets is 8 times that of the original, with a total of 16,000 pictures. Examples of time-frequency pictures under different expansion methods are shown in Figure 3. In order to ensure a balanced sample size, 2000 images of different categories were randomly selected from each type of samples, totaling 10,000 images. According to the ratio of 8:2, they are divided into a training set (8000 pictures) and test set (2000 pictures).



a.Original image b.Rotation c. Flip horizontal d. Flip vertical

e.Brightness f. Saturation g. Contrast h. Gaussian blur

Figure 3. Example of machine tool OS images after data augmentation.

3.3. Recognition Model Establishment

3.3.1. Network Construction

The AlexNet Convolutional Neural Network (ACNN) is an improved algorithm of the traditional Convolutional Neural Network, which uses training techniques such as the ReLU activation function, overlapping pooling, dual GPU training, data augmentation and dropout, and satisfies the requirements of power data processing [40]. The ACNN was firstly proposed in the Large-scale Visual Recognition Challenge (ILSVRC 2012) [41], which is a deep convolutional neural network based on the classic LeNet5 and the traditional BP neural network [42], with the network construction shown in Figure 3.

Now, we are ready to describe the overall architecture of our ACNN. As shown in Figure 4, the convolutional layer has a total of five layers (C1-C5), and the fully connected layer has a total of three layers (FC1-FC3).

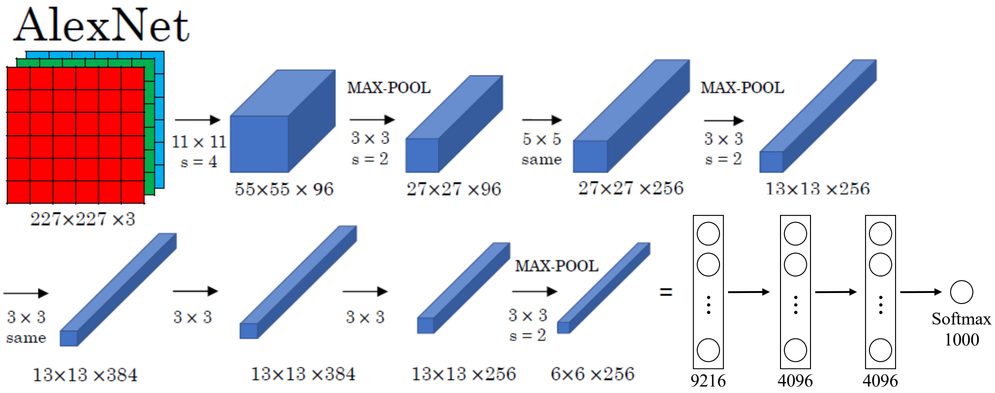


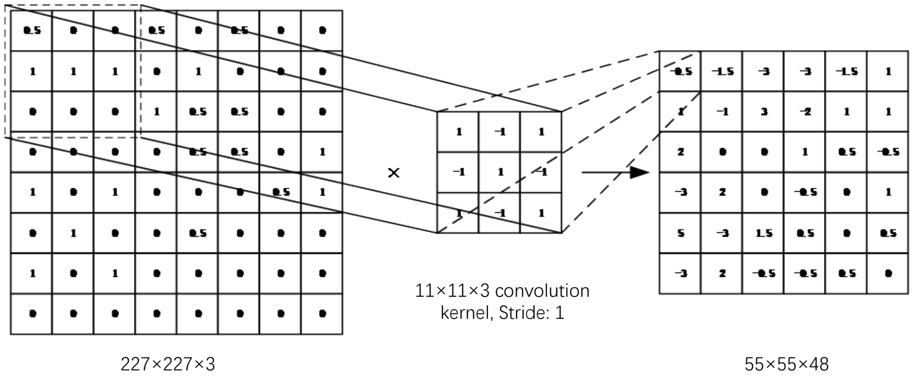
Figure 4. Overall framework for the improved network.

The input data of the first layer (C1) is the original 227 × 227 × 3 image, which is convolved by a 11 × 11 × 3 convolution kernel, and the moving step size is 4 pixels. (This is the receptive field of adjacent neurons in the kernel map [43] distance between centers). The second convolutional layer (C2) takes as input the (response normalization and pooling) output of the first convolutional layer and filters it with 256 kernels of size 5 × 5 × 48. The input data of the second layer is the pixel layer output by the first layer. In order to facilitate subsequent processing, the left and right sides and the upper and lower sides of each pixel layer should be filled with 2 pixels [44]; Operations are then performed on different GPUs. Each set of pixel data is convolved with 5 × 5 × 48 convolution kernels, and a total of 256 kernels of size 5 × 5 × 48 are used to filter them. The third, fourth and fifth convolutional layers (C4, C5) are connected to each other without any intermediate pooling or normalization layers. The third convolutional layer has 384 kernels of size 3 × 3 × 256 connected to the (normalized, pooled) outputs of the second convolutional layer. The fourth convolutional layer has 384 kernels of size 3 ×3 × 192, and the fifth convolutional layer has 256 kernels of size 3 × 3 × 192. Fully connected layers have 4096 neurons each.

The last three layers are the fully connected layer (FC), which is employed to expand the multi-dimensional matrix data generated by the convolution of the last convolution layer into one dimension, and predict each category.

(1) Convolutional layer

The main goal of the convolutional layer is to extract features from the input image. On the convolutional layer, multiple convolution kernels are used to convolve the input image, and after adding paranoia, a series of output feature maps can be obtained through a nonlinear activation function. This layer has characteristics of local connection and weight sharing, both of which reduce the complexity of the model and reduce the number of parameters. Among them, taking the convolution of a single GPU of the C1 convolution layer as an example, the convolution process of AlexNet is shown in Figure 5 and Equation (4).



0.5	0	0	0.5	0	0.5	0	0
1	1	1	0	1	0	0	0
0	0	0	1	0.5	0.5	0	0
0	0	0	0	0.5	0.5	0	1
1	0	1	0	0	0	0.5	1
0	1	0	0	0.5	0	0	0
1	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0

1	-1	1
-1	1	-1
1	-1	1

-0.5	-1.5	-3	-3	-1.5	1
1	-1	3	-2	1	1
2	0	0	1	0.5	-0.5
-3	2	0	-0.5	0	1
5	-3	1.5	0.5	0	0.5
-3	2	-0.5	-0.5	0.5	0

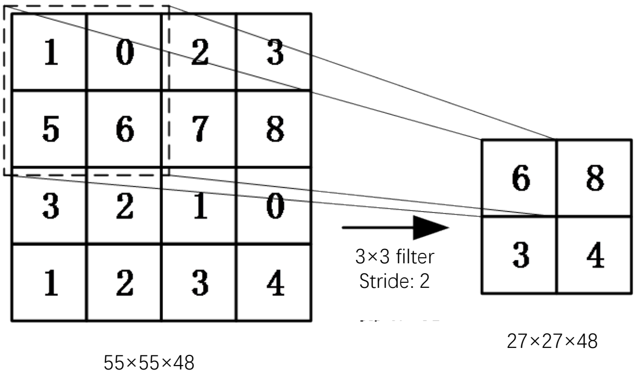
Figure 5. The process of Convolution.

$$X_j^l = f\left(\sum_{i \in M_j} X_i^{l-1} \cdot w_{ij}^l + b_j^l\right), \tag{4}$$

where, X_j^l is the element of the second layer; M_j is the fourth convolution area of the *j*-layer feature map; X_i^{l-1} is the element in it; w_{ij}^l is the weight matrix corresponding to the convolution kernel; b_j^l is the paranoid term. *f* is the activation function ReLU.

(2) Pooling layer

The main function of the pooling layer is to reduce the dimension of the feature map while maintaining the invariance of the feature scale to a certain extent. Pooling layers reduce the number of parameters and computation by gradually reducing the size of the representation space to control overfitting. Pooling layers usually take a convolutional layer as input. The pooling process is shown in Figure 6 and Equation (5).



1	0	2	3
5	6	7	8
3	2	1	0
1	2	3	4

6	8
3	4

Figure 6. The process of Pooling.

$$X_j^l = f(\beta_j^l down(X_i^{l-1}) + b_j^l), \tag{5}$$

where, *down()* is the subsampling function; β_j^l is the weight of the *j*-feature map of the *l*-layer; b_j^l is the bias of the *j*-feature map of the *l*-layer.

(3) Fully Connected layer

After the input image is alternately propagated through multiple convolutional layers and pooling layers, the fully connected layer network is used to classify the extracted features. In the fully connected layer, the one-dimensional feature vectors expanded by all feature maps at the input are obtained by weighted summation and passing through the activation function. The calculation process is shown in Equation (6).

$$y^k = f(w^k x^{k-1} + b^k), \tag{6}$$

where, k is the serial number of the network layer; y^k is the output of the fully connected layer; x^{k-1} is the expanded one-dimensional feature vector; w^b is the weight coefficient; f is the activation function.

(4) Regularization

Due to the high complexity of deep learning models, large-scale training data are critical for model robustness. However, in the tool wear monitoring problem, it is difficult to obtain large-scale training data. Therefore, this paper introduces ReLU [45], LRN [46], and Dropout [47] into the ACNN model training to regularize the network.

- Nonlinear activation function ReLU. Its gradient descent is always 1, which can effectively prevent the problem of gradient disappearance and shorten the training time. This operation can be represented by Equation (7).

$$f(x) = \max(0, x), \tag{7}$$

- Local Response Normalization. Local response normalization (LRN) creates a competition mechanism for the activity of local neurons, rendering the response values increasingly large, thus inhibiting other neurons with smaller responses, enhancing the generalizability of the model.
- Dropout. Randomly deleting some neurons through a defined probability, while keeping the number of neurons in the input layer and output layer unchanged, and update the parameters. Repeat these operations in the next iteration until the end of training. Dropout can effectively prevent overfitting of neural networks.

3.3.2. Network Training and Parameter Adjustment

The parameters and regression layer feature weights in the ACNN model need to be learned through model training, and thus the model needs to be trained through training data to obtain the optimal parameters in the model. The difference between the predicted value and the true value for a particular sample is defined as loss. Here, the authors choose to use an Adaptive moment estimation (Adam) [48] to minimize the SoftMax categorical cross-entropy loss function, the loss function is defined as Equation (8).

$$Loss = -\frac{1}{N}\sum_{i=1}^{N} y'_{i1} ln y_{i1} + y'_{i2} ln y_{i2} + \ldots + y'_{iM} ln y_{iM}, \tag{8}$$

where, N represents the number of samples; M represents the number of classes; y' represents the predicted output value; y displays the actual value.

Model predictions where the model has been tested against a given test dataset by collecting power signals. During testing, there are some hyperparameters for stochastic gradient descent for training convolutional neural networks. In the proposed ACNN model, two main hyperparameters are investigated to improve the recognition accuracy.

- The initial learning rate [49]. The initial learning rate controls how well we adjust the network weights and error convergence based on the gradient of the loss.
- Batch size [50]. The stochastic gradient descent method is used in training convolutional neural network, and its batch size affects computer memory utilization and training oscillation which represents the number of training examples in a single batch. The batch size limits the number of training samples before each weight update.

3.4. Model Validation

The ACNN is essentially a deep learning-based classifier, the classified performance of which could be verified with the data classification validation models [51]. In this paper, the confusion matrix is used to evaluate the classified performance of the ACNN.

For the classification task M, the number of samples is $N(N = 2000)$, $C = \{C_1, C_2, \ldots, C_L\}$ is the classifier set (i.e., ACNN), and the $L(L = 1)$ classifiers are used to test in the samples N respectively. The confusion matrix $C_k(k = 1, 2, \ldots, L)$ of each classifier is obtained, and the confusion matrix of the kth classifier C_k is shown in Equation (9):

$$C_k = \begin{bmatrix} N_{1,1}^k & N_{1,2}^k & \cdots & N_{1,M}^k \\ N_{2,1}^k & N_{2,2}^k & \cdots & N_{2,M}^k \\ \vdots & \vdots & N_{i,j}^k & \vdots \\ N_{M,1}^k & N_{M,2}^k & \cdots & N_{M,M}^k \end{bmatrix}, \tag{9}$$

where, the elements in the i-th row and the j-th column represent the number of the i-th class identified as the j-th class by the classifier C_k in the sample. If $i = j$, it means that the classifier can correctly recognize the number of samples, so the diagonal elements represent the classifier C_k.

The number of correct classifications, and the off-diagonal elements represent the number of misclassifications by the classifier C_k. If, ideally, both the recall and precision of the classifier are 100%, then the off-diagonal elements of the confusion matrix are all 0, and only the diagonal elements are non-zero.

In this study, the accuracy rate (Accuracy, %) was used to evaluate the performance of all different OS classification models, and the misclassification of five OS was analyzed by confusion matrix. The accuracy is the ratio of the number of correctly classified samples to the total number of samples, which is calculated as Equation (10)

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \times 100\%, \tag{10}$$

where, *TP*, *FP*, *FN*, and *TN* are the statistics of the classification of different OS by the classification model in the confusion matrix, respectively. Among them, *TP* is the number of samples judged as positive in the positive class, *FP* is the number of samples judged as positive in the negative class, *FN* is the number of samples in the positive class judged as the negative class, and *TN* is the number of samples in the negative class judged as the negative class. When performing classification tasks, the number of real categories of samples to be predicted is regarded as the number of positive samples, and the sum of all other categories is the number of negative samples.

4. Case Study

4.1. Experimental Conditions

In this paper, a face milling experiment with the milling machine XK713 is designed to verify the proposed model and approach. In this case, the WT1800 power tester is used to measure three-phase current signal during machining, and the different machining parameters (cutting speed n, feed amount f_v and back cut amount a_p) are also set up to illustrate the generalization ability of the proposed approach.

Test environment for ACNN model training: hardware includes Intel(R) Core(TM) i7-6500U CPU@2.5GHz processor with 20 GB of memory and an NVIDIA GeForce RTX 940M graphics card. Software includes operating system Ubantu22.04 (64-bit), programming language Python3.9, Deep learning framework Pytorch1.12.0, general computing architecture CUDA10.2 and GPU acceleration library CUDNN8.5.0. The main processing equipment and parameters are shown in Figure 7 and Table 1, respectively.

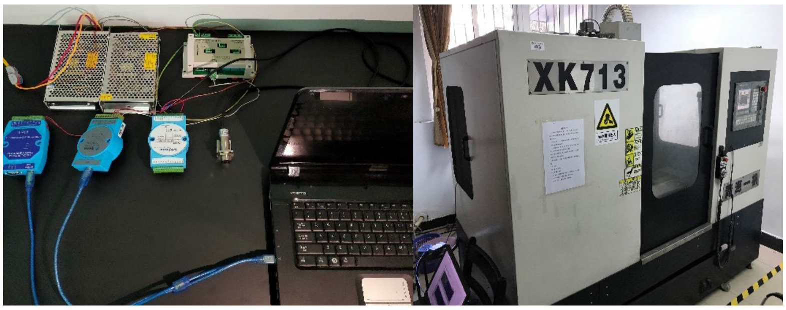


Figure 7. Equipment used at the experimental site.

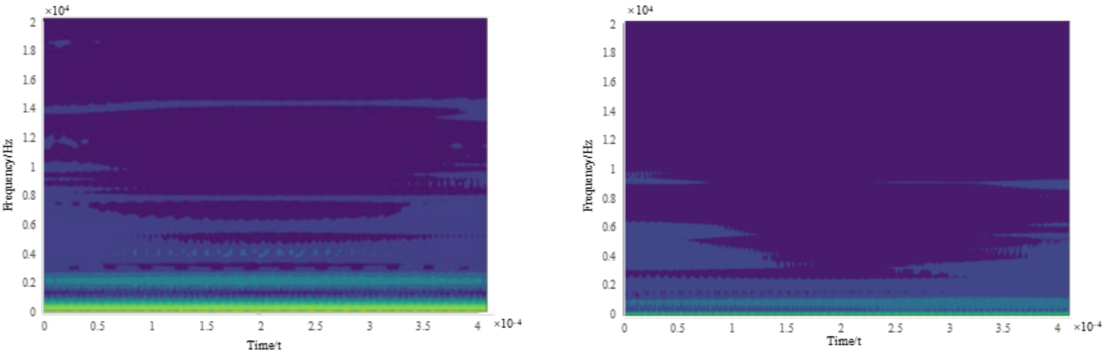
Table 1. Main parameters of cutting activity.

Activities	Tool Diameter	n /(r/min)	f_v /(mm/a)	a_p /(mm)	$L_{cutting}$ /(mm)	L_{air} /(mm)
1	12	800	225	1.5	600	200
2	12	450	150	6	464	32
3	12	450	150	2	600	45
4	12	950	262.5	0.2	582	56
5	12	950	262.5	6	324	28
6	12	600	187.5	2	159	72
7	12	600	187.5	2	159	72

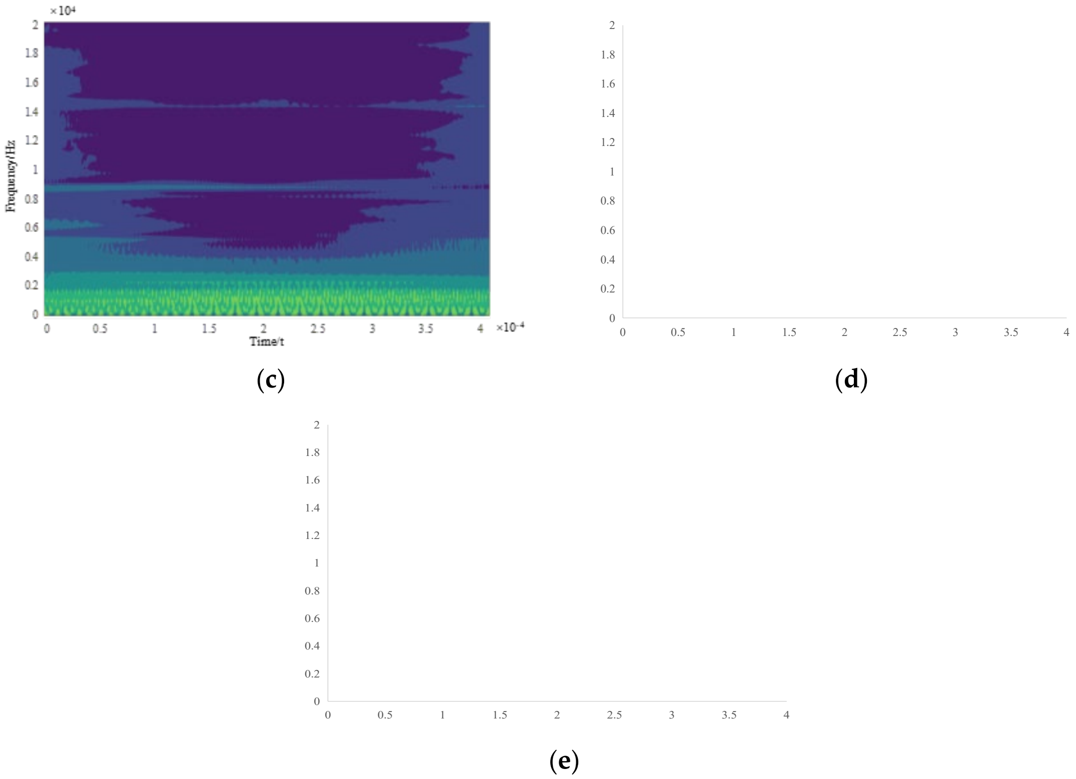
4.2. Results and Discussion

4.2.1. Results

With the proposed WT approach, the time-frequency analysis results for different OS current signals of the XK713 are obtained, and the time-frequency images for each state are shown in Figure 8a–e, respectively. It can be seen that there are obvious differences in the time-frequency images of the current signals for the five OS, and the time-frequency images can be used to recognize the OS of XK713.



(**a**) (**b**)

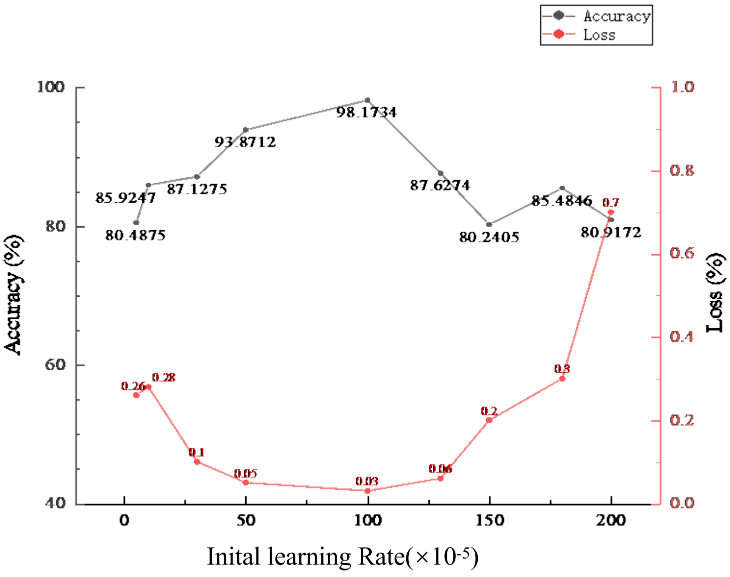


(**c**) (**d**)

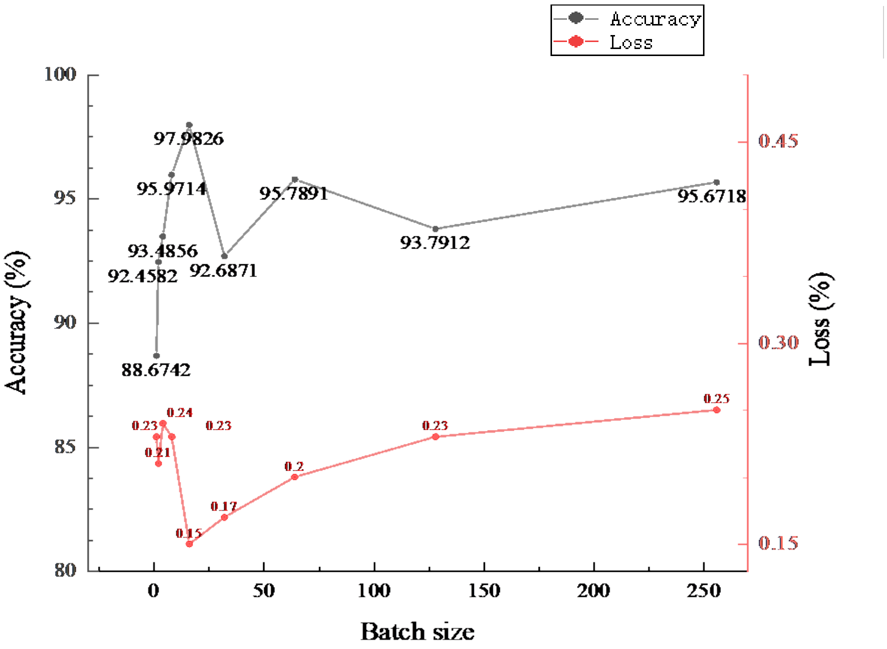
(**e**)

Figure 8. The variance curve of the detail signal D3 obtained by wavelet decomposition of the W-phase electric current signal. (**a**) S1 state wavelet decomposition time-frequency image; (**b**) S2 state wavelet decomposition time-frequency image; (**c**) S3 state wavelet decomposition time-frequency image; (**d**) S4 state wavelet decomposition time-frequency image; (**e**) S5 state wavelet decomposition time-frequency image.

To study the tuning method of two hyperparameters, namely initial learning rate and batch size, the Adam optimizer was used to set different initial learning rates and batch sizes to train the model, and the effects of different parameters on the accuracy of the model were compared and analyzed. The variation trend of the accuracy of the test set with the parameters is shown in Figure 9a,b.



(**a**)

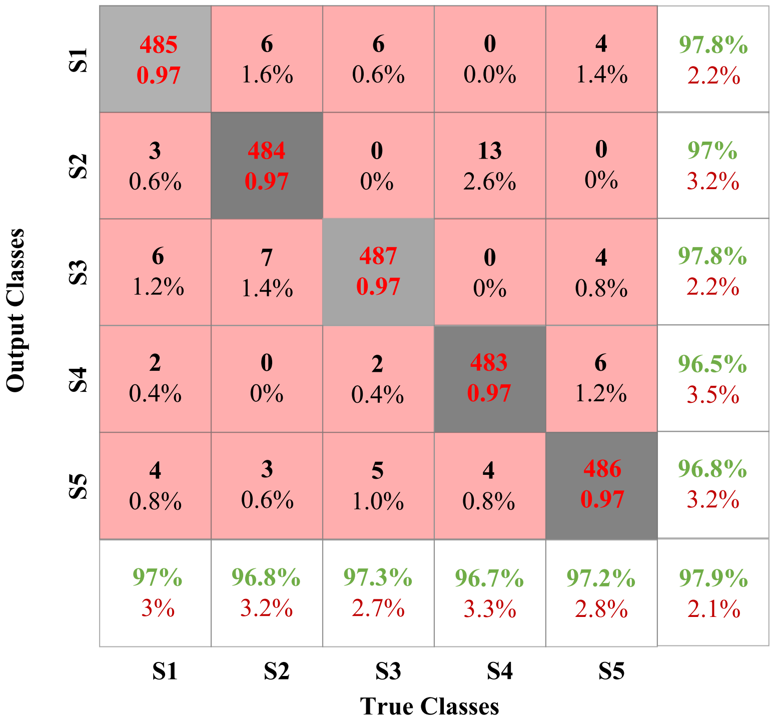


(**b**)

Figure 9. The effect of hyperparameters on accuracy and loss. (**a**) The effect of initial learning rate value on accuracy value and loss; (**b**) The effect of batch size value on accuracy and loss.

Considering the influence of the experimental environment, blindly pursuing high batch size and low initial learning rate cannot achieve the expected effect. When the learning rate is less than 0.0005 or greater than 0.002, the loss value has an upward trend. When the learning rate is set within 0.001, the ACNN network has the lowest loss value with high accuracy. When the batch size value is greater than 64 or less than 8, the accuracy rate has a downward trend; when the batch size is set to 16, ACNN has the highest accuracy rate and the lowest loss value. Therefore, the optimized values for the initial learning rate and batch size are 0.001 and 16.

After selecting the optimal hyperparameters, the classification performance of the model is evaluated using the confusion matrix, and the results are shown in Figure 10.



Output Classes / True Classes	S1	S2	S3	S4	S5	
S1	485 0.97	6 1.6%	6 0.6%	0 0.0%	4 1.4%	97.8% 2.2%
S2	3 0.6%	484 0.97	0 0%	13 2.6%	0 0%	97% 3.2%
S3	6 1.2%	7 1.4%	487 0.97	0 0%	4 0.8%	97.8% 2.2%
S4	2 0.4%	0 0%	2 0.4%	483 0.97	6 1.2%	96.5% 3.5%
S5	4 0.8%	3 0.6%	5 1.0%	4 0.8%	486 0.97	96.8% 3.2%
	97% 3%	96.8% 3.2%	97.3% 2.7%	96.7% 3.3%	97.2% 2.8%	97.9% 2.1%

Figure 10. Confusion matrix of the proposed approach.

In the confusion matrix, the rows and columns of the confusion matrix represent the True label and Output label of each condition [52]. Figure 10 shows the confusion matrix of the test sample used in the improved ACNN, giving detailed classification results.

As shown in Figure 7, the overall recognition accuracy of S3 is the highest, and the highest error rate is only 1% (only 1% of S5 is recognized as S3). The reason may be that the current fluctuates violently when the machine tool is turned on, and its characteristics are easier to identify. However, the overall recognition accuracy of S4 is relatively low, indicating that ACNN misclassifies S4 the most. Even 2.6% of S4 were misidentified as S2, the highest among all experiments. It is foreseeable that the current fluctuations of S4 and S2 are not clear, the gap between the peak and the trough is not large, and can easily be erroneously classified. In addition, S1 has the most balanced recognition accuracy, with the highest recognition error rate (S3) and the lowest recognition error rate (S4) differing by 0.8%. In general, the recognition accuracy of the five states can reach about 97%, the highest can reach 97.8%, the lowest can reach 96.5%, and the peak difference is only 1.3%. It can be seen that the method combining WT and ACNN not only has high classification accuracy, but also has relatively stable classification performance.

Furthermore, by further analyzing the misclassification of different machine tool operating system datasets, this study shows that the color and texture features of different OS are important basis in the classification process. Misclassification is likely to occur when there is noise. The misclassified samples are mostly images with high brightness, low saturation and added Gaussian blur. Its complex background interferes with the characteristics of the image to some extent, which affects the accurate identification of the sample.

4.2.2. Discussion

To further illustrate the superiority of the proposed model, this paper uses the following four common models to analyze from the perspective of recognition accuracy and computational cost: Linear Regression (LR), Backpropagation (BP), LeNet-5 and Residual Network (ResNet-18). The first three models bear fewer network layers and are more commonly used early classification models, and the fourth model is currently a more popular classification model. To investigate the performance of the proposed feature extraction method, three typical feature extraction methods were selected for comparison: engineering methods, statistical methods, and principal component analysis (PCA). For computational load, there are two methods of computational load that can be used to analyze ACNN models:

- T_1 : Time spent on modeling during training;
- T_2 : The time taken to generate the sample.

Each model was run 20 times, using the same initial learning rate, batch size, and training set, and the final average was taken as the result. Table 2 shows the corresponding results for the five classifiers. It can be seen that the maximum classification accuracy rates are 85.17%, 89.17%, 91.56%, 97.89% and 98.14%, respectively.

As shown in Table 2, compared with the linear model, the excellent performance of the deep learning model in model recognition shows that the feature extraction of the original data by the deep learning model can mine deeper and more comprehensive feature information, which proves the feasibility and effectiveness of Deep Learning Models. With the increase in the network structure, the accuracy of ACNN is much higher than that of LeNet-5, and it is not much inferior to ResNet-18. On the contrary, when calculating the load, the time spent by ResNet-18 is multiplied compared to that of ACNN, and the impact on the server and the actual production process is self-evident.

Table 2. Accuracy of different models and time spent modeling and generating samples.

Type of Comparison		Linear Regression	BP	LeNet-5	ACNN	ResNet-18
Training times		20	20	20	20	20
Initial learning rate		/	/	0.01	0.01	0.01
Accuracy (%)		85.17	89.17	91.56	97.89	98.14
T_1	EST	0.016	0.055	0.028	0.040	0.184
	STAT	20.357	28.387	30.489	39.224	111.874
	PCA	16.872	17.5780	20.784	33.874	109.477
T_2	EST	0.014	0.041	0.046	0.102	0.544
	STAT	0.014	0.058	0.027	0.093	0.578
	PCA	0.026	0.034	0.029	0.031	0.129

It can also be seen from the table that LR requires the least computational time to develop the model in the milling machine. In contrast, the proposed ACNN is computationally expensive because ACNN spends a lot of time on convolution and pooling operations. ResNet-18 obviously increases the computational cost significantly after adopting the residual structure. Average computation time for generating samples. The longest one is ResNet-18, because a lot of time is spent on updating features. Unexpectedly, ACNN is almost the same as BP and LeNet-5 when adopting the PCA feature lifting method

By comparison, it can be found that the fitting effect of the ACNN model is significantly better than that of other classification models. The main advantages of proposed model are:

- The training parameters are adjusted. Because the adjusted model network has a suitable batch size, it can aggregate faster. In particular, the complex classification of the energy consumption of machine tools is especially notable. The data collection time is short, and easily affected by the processing environment;
- Feature extraction techniques such as wavelet decomposition are combined with deep learning networks. The performance went well on small training samples, the network is very sensitive to the characteristics of the data, and it is not prone to loss and overfitting;

- The structure of the model. The ACNN model facilitates real-time data processing without human intervention. Moreover, the operation speed is improved by using ReLU and multiple CPUs. At the same time, techniques such as overlapping pooling, data gain, and dropout are used to improve the operational accuracy of ACNN and reduce overfitting.

5. Conclusions

In this paper, an ACNN-based current data-driven identification method of a machine tool operating system is proposed. The identification of the operating system is a precondition for the fault diagnosis of the machine tool, and is of great significance to the fault diagnosis of the machine tool. This study only requires raw current data to identify the machine tool operating system and mine sensitive energy consumption features from the data. Compared with the traditional threshold method and experimental method, the generalization is stronger, and the requirements for the scene environment and artificial feature extraction are not high. In addition, this paper extracts the temporal features of the data through WT, which reduces the noise in the process of converting the data to grayscale images. While meeting the input requirements of ACNN, more accurate feature information can be included. Finally, a series of experiments were carried out. The results show that compared with linear regression, BP, LeNet-5 and RESNET-18, classifiers, ACNN performs better than the above-mentioned classifiers in most cases. Furthermore, it has faster recognition speed and higher recognition accuracy.

The proposed ACNN provides an efficient, simple and fast method to determine the operating system of a machine tool. The method has certain guiding significance for reducing maintenance time, improving system reliability and ensuring the safe operation of machine tool systems. In the real-world production environment, the collected machine tool current data are becoming increasingly complex. At present, the model has only been verified in milling processing, and its application in turning and other processing can be considered in the follow-up research. This can be aided by increasing the scale of model training data to further improve the accuracy of the mode, improving the test environment, and adopting more advanced AI algorithms to improve the recognition performance.

Author Contributions: Conceptualization, Methodology, Software and Validation, W.Y., C.L. and Y.L.; Formal analysis, Investigation and Resources, X.Z.; Data curation, W.Y., C.L.; Writing—original draft preparation, W.Y. and C.L.; Writing—review and editing, Y.L.; Visualization, W.Y., C.L. and X.Z.; Supervision, Y.L. and H.Z.; Project administration, W.Y. All authors have read and agreed to the published version of the manuscript.

Funding: The research was funded by the National Natural Science Foundation of China grant number 51975432, and the China Scholarship Council for Visiting Scholars grant number 202008420116.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: The datasets used and/or analyzed during the current study are available from the corresponding author on reasonable request.

Acknowledgments: The research was funded by the National Natural Science Foundation of China grant number 51975432, and the China Scholarship Council for Visiting Scholars grant number 202008420116. Thanks here.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Li, Y.; He, Y.; Wang, Y.; Wang, Y.; Yan, P.; Lin, S. A modeling method for hybrid energy behaviors in flexible machining systems. *Energy* **2015**, *86*, 164–74.
2. He, Y.; Liu, B.; Zhang, X.; Gao, H.; Liu, X. A modeling method of task−oriented energy consumption for machining manufacturing system. *J. Clean. Prod.* **2012**, *23*, 167–174. https://doi.org/10.1016/j.jclepro.2011.10.033.
3. Liu, H.; Yan, P.; Zhou, Q.; Xin, Y.; Zhang, Z.K.; Pan, J.C. Knowledge Expansion Method for Machine Tool Fault Diagnosis Based on Semantic Web. *Comput. Integr. Manuf. Syst.* **2020**, *26*, 609–622. https://doi.org/10.13196/j.cims.2020.03.004.
4. Fu, Z.H. Research on the Fault Diagnosis System of CNC Manufacturing Unit based on Multi−Sensor Information Fusion. Master`s Thesis, University of Electronic Science and Technology, Chengdu, China, 2014.
5. Gutowski, T.; Branham, M.; Dahmus, J.; Jones, A.; Thiriez, A.; Sekulic, D.P. Thermodynamic analysis of resources used in manufacturing processes. *Environ. Sci. Technol.* **2009**, *43*, 1584–1590.
6. Liu, S.; Liu, F.; Hu, S.; Yin, Z. Energy survey of machine tools: Separating power information of the main transmission system during machining process. *J. Adv. Mech. Des. Syst. Manuf.* **2012**, *6*, 445–455.
7. Wu, J.; Yu, G.; Gao, Y.; Wang, L. Mechatronics modeling and vibration analysis of a 2−DOF parallel manipulator in a 5−DOF hybrid machine tool. *Mech. Mach. Theory* **2018**, *121*, 430–445. https://doi.org/10.1016/j.mechmachtheory.2017.10.023.
8. Li, H.; Wang, W.; Huang, P.; Du, L.; Zhang, X. Research on the state prediction method of CNC machine tools based on multi−dimensional time series. *Eng. Sci. Technol.* **2018**, *50*, 187–195. https://doi.org/10.15961/j.jsuese. 201700435.
9. Liu, P.; Zhang, Z.; Wang, X.; Li, X.; Wang, X.V.; Tuo, J. A generalized method for the inherent energy performance modeling of machine tools. *J. Manuf. Syst.* **2021**, *61*, 406–422.
10. Liu, G.; Liu, F.; Liu, P.; Wu, P. On−line detection method and detection system for multi−source energy consumption status of machine tools. *Comput. Integr. Manuf. Syst.* **2016**, *22*, 1550–1557. https://doi.org/10.13196/j.cims.2016.06.017.
11. Zhang, Y.Z.; Mu, L.M.; Shen, G.X.; Han, C.Y. Fault diagnosis strategy of CNC machine tools based on cascading failure. *J. Intell. Manuf.* **2019**, *30*, 2193–2202. https://doi.org/10.1007/s10845−017−1382−7.
12. Oliver, I.A.; Paul, X. Evaluating the use phase energy requirements of a machine tool system. *J. Clean. Prod.* **2010**, *19*, 699–711. https://doi.org/10.1016/j.jclepro.2010.10.010.
13. Lv, J.X.; Tang, R.Z.; Wang, C.J.; Jia, S. An investigation into reducing the spindle acceleration energy consumption of machine tools. *J. Clean. Prod.* **2016**, *143*, 794–803 https://doi.org/10.1016/j.jclepro.2016.12.045.
14. Martin, K.; Jiri, V.; Jan, S. Analytical approach to establishment of predictive models of power consumption of machine tools' auxiliary units. *J. Clean. Prod.* **2016**, *137*, 361–369. https://doi.org/10.1016/j.jclepro.2016.07.092.
15. Reza, I.A.; Stephen, T.N.; Aydin, N. A mechanistic model of energy consumption in milling. *Int. J. Prod. Res.* **2018**, *56*, 642–659. https://doi.org/10.1080/00207543.2017.1404160.
16. Hu, S.H.; Liu, F.; He, Y.; Hu, T. No−load energy parameter characteristics of computerized numerical control machine tool main transmission system. *Comput. Integr. Manuf. Syst.* **2012**, *18*, 326–331.
17. Wu, J.; Zhang, B.; Wang, L.; Yu, G. An iterative learning method for realizing accurate dynamic feedforward control of an industrial hybrid robot. *Sci. China: Tech. Sci.* **2021**, *64*, 1177–1188.
18. Benjamin, M.; Philipp, D.; Martin, P.; Robert, H.S. Modelling Machine Tools using Structure Integrated Sensors for Fast Calibration. *J. Manuf. Mater. Processing* **2018**, *2*, 14. https://doi.org/10.3390/jmmp2010014.
19. Liu, P.; Liu, S.; Liu, F. Calculating method for additional load loss coefficient of spindle system of CNC machine tools. *Chin. J. Mech. Eng.* **2016**, *52*, 121–128.
20. Wei, Y.; Hua, Z.; Zhi−Gang, J.; Hon, K.K.B. A new multi−source and dynamic energy modeling method for machine tools. *Int. J. Adv. Manuf. Technol.* **2018**, *95*, 4485–4495. https://doi.org/10.1007/s00170−017−1545−x.
21. Liu, F.; Liu, S. Period energy model of electromechanical main drive system during service of machine tool. *Chin. J. Mech. Eng.* **2012**, *48*, 132–140. https://doi.org/10.3901/JME.2012.21.132.
22. Abele, E.; Altintas, Y.; Brecher, C. Machine tool spindle units. *CIRP Ann.* **2010**, *59*, 781–802.
23. Brecher, C.; Jasper, D.; Fey, M. Analysis of new, energy−efficient hydraulic unit for machine tools. *Int. J. Precis. Eng. Manuf. Green Technol.* **2017**, *4*, 5–11.
24. Shabi, L.; Weber, J.; Weber, J. Analysis of the energy consumption of fluidic systems in machine tools. *Procedia Cirp* **2017**, *63*, 573–579.
25. He, Y.; Lin, S.L.; Wang, Y.L.; Li. Y.F.; Wang. L.X. Dynamic energy consumption modeling and simulation method for CNC machine tools with multiple energy sources. *Chin. J. Mech. Eng.* **2015**, *51*, 123–132. https://doi.org/10.3901/JME.2015.11. 123.
26. Jia, S.; Tang, R.; Lv, J.X. Therblig−based energy demand modeling methodology of machining process to support intelligent manufacturing. *J. Intell. Manuf.* **2014**, *25*, 913–931. https://doi.org/10.1007/s10845−012−0723−9.
27. Nayana, B.; Geethanjali, P. Analysis of statistical time−domain features effectiveness in recognition of bearing faults from vibration signal. *IEEE Sens. J.* **2017**, *17*, 5618–5625.
28. Liao, Y.; Sun, P.; Wang, B.; Qu, L. Extraction of repetitive transients with frequency domain multipoint kurtosis for bearing fault diagnosis. *Meas. Sci. Technol.* **2018**, *29*, 055012.
29. Zhang, X.; Liu, Z.; Wang, J.; Wang, J. Time−frequency analysis for bearing fault diagnosis using multiple Q−factor Gabor wavelets. *ISA Trans.* **2019**, *87*, 225–234.
30. Gao, H.; Liang, L.; Chen, X.; Xu, G. Feature extraction and recognition for rolling element bearing fault utilizing short−time Fourier transform and non− Negative matrix factorization. *Chin. Int. J. Mech. Eng. Educ.* **2015**, *28*, 96–105.

31. Babu, G.S.; Zhao, P.; Li, X.−L. Deep convolutional neural network based regression approach for estimation of remaining useful life. In *International Conference on Database Systems for Advanced Applications*; Springer: New York, NY, USA, 2016; pp. 214–228.
32. Shin, H.−C.; Roth, H.R.; Gao, M.; Lu, L.; Xu, Z.; Nogues, I.; Yao, J.; Mollura, D.; Summers, R.M. Deep convolutional neural networks for computer−aided detection: CNN architectures, dataset characteristics and transfer learning. *IEEE Trans. Med. Imaging* **2016**, *35*, 1285–1298.
33. Wan, L.J.; Chen, Y.W.; Li, H.Y.; Li, C.Y. Rolling−Element Bearing Fault Diagnosis Using Improved LeNet−5 Network. *Sensors* **2020**, *20*, 1693. https://doi.org/10.3390/s20061693.
34. Shi, Y.; Zeng, S.K.; Li, Y.; Liu, H.Q. Recognition of effective machining state of machine tool based on deep convolutional neural network. *Mil. Civ. Technol. Prod.* **2020**, *11*, 54–58.
35. Ghulanavar, R.; Dama, K.K.; Jagadeesh, A. Diagnosis of faulty gears by modified AlexNet and improved grasshopper optimization algorithm (IGOA). *J. Mech. Sci. Technol.* **2020**, *34*, 4173–4182. https://doi.org/10.1007/s12206−020−0909−6.
36. Tabernik, D.; Šela, S.; Skvarč, J.; Skočaj, D. Segmentation−based deep−learning approach for surface−defect detection. *J. Intell. Manuf.* **2020**, *31*, 759–776.
37. Xing, L.; Wang, H.; Tang, X.; Wang, G. Dual−branch convolutional neural network fault diagnosis method considering fault location and damage degree of rolling bearing. *Sci. Technol. Eng.* **2022**, *22*, 1441–1448; ISSN 1671−1815.2022.04.018. https://doi.org/10.3969/j.
38. Coelho, I.M.; Coelho, V.N.; Luz, E.; Ochi, L.S.; Guimarães, F.G.; Rios, E. A GPU deep learning metaheuristic based model for time series forecasting. *Appl Energy* **2017**, *201*, 412–418.
39. Zhang, T.H.; Flores, M.; Huang, Y.F. ES−ARCNN: Predicting enhancer strength by using data augmentation and residual convolutional neural network. *Anal. Biochem. Int. J. Anal. Prep. Methods* **2021**, *618*, 114120. https://doi.org/10.1016/j.ab.2021.114120.
40. Dhar, P.; Dutta, S.; Mukherjee, V. Cross−wavelet assisted convolution neural network (AlexNet) approach for phonocardiogram signals classification. *Biomed. Signal Process. Control* **2020**, *63*, 102142. https://doi.org/10.1016/J.BSPC.2020.102142.
41. Alex, K.S.; Ilya, S.E.; Geoffrey, E.H. ImageNet Classification with Deep Convolutional Neural Networks. In Proceedings of the Advances in Neural Information Processing Systems 25, 26th Annual Conference on Neural Information Processing Systems 2012, Lake Tahoe, NV, USA, 3–6 December 2012; Neural Information Processing Systems; Volume 2, pp. 1097–1105
42. Zhu, L.; Chang, W.K. Application of Deep Convolutional Neural Networks in Attention−Deficit/Hyperactivity Disorder Classification: Data Augmentation and Convolutional Neural Network Transfer Learning. *J. Med. Imaging Health Inform.* **2019**, *9*, 1717–1724. https://doi.org/10.1166/jmihi.2019.2843.
43. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. *Commun. ACM* **2017**, *60*, 84–90. https://doi.org/10.1145/3065386.
44. Singh, P.; Verma, V.K.; Rai, P.; Namboodiri, V.P. Acceleration of Deep Convolutional Neural Networks Using Adaptive Filter Pruning. *IEEE J. Sel. Top. Signal Processing* **2020**, *14*, 838–847. https://doi.org/10.1109/JSTSP.2020.2992390.
45. Xu, Y.; Wang, H.; Liu, X. An improved multi−branch residual network based on random multiplier and adaptive cosine learning rate method. *J. Vis. Commun. Image Represent.* **2019**, *59*, 363–370. https://doi.org/10.1016/j.jvcir.2019.01.030.
46. Zhang, P. Research on Ensemble Method of Classifier Selection Based on Confusion Matrix. Master`s Thesis, Henan University of Technology, Henan, China, 2016.
47. Chen, C.; Liu, Y.; Sun, X.; Di Cairano−Gilfedder, C.; Titmus, S. Automobile Maintenance Prediction Using Deep Learning with GIS Data. *Procedia CIRP* **2019**, *81*, 447–452.
48. Hosamelen, O.A.A.; Asoke, K.N. Connected Components−based Colour Image Representations of Vibrations for a Two−stage Fault Diagnosis of Roller Bearings Using Convolutional Neural Networks. *Chin. J. Mech. Eng.* **2021**, *34*, 73–93.
49. Abdeljaber, O.; Avci, O.; Kiranyaz, M.S.; Boashash, B.; Sodano, H.; Inman, D.J. 1−D CNNs for structural damage detection: Verification on a structural health monitoring benchmark data. *Neurocomputing* **2018**, *275,* 1308–1317. https://doi.org/10.1016/j.neucom.2017.09.069.
50. Liu, J.; Mi, X.W.; Li, Y.F. Smart deep learning based wind speed prediction model using wavelet packet decomposition, convolutional neural network and convolutional long short term memory network. *Energy Convers. Manag.* **2018**, *166*, 120–131. https://doi.org/10.1016/j.enconman.2018.04.021.
51. Rachuri, S.; Park, J.; Law, K.; Dornfeld, D.; Helu, M.; Bhinge, R. Toward a generalized energy prediction model for machine tools. *J. Manuf. Sci. Eng.−Trans. ASME* **2016**, *139*, 041013. https://doi.org/10.1115/1.4034933.
52. Singh, A.; Rajan, P.; Bhavsar, A. SVD−based redundancy removal in 1−D CNNs for acoustic scene classification. *Pattern Recognit. Lett.* **2020**, *131*, 383–389. https://doi.org/10.1016/j.patrec.2020.02.004.