

# Prediction of the Remaining Useful Life of Cutting Tool using the Hurst Exponent and CNN-LSTM

Zhang, X., Lu, X., Li, W. & Wang, S.

Author post-print (accepted) deposited by Coventry University's Repository

**Original citation & hyperlink:**

Zhang, X, Lu, X, Li, W & Wang, S 2021, 'Prediction of the Remaining Useful Life of Cutting Tool using the Hurst Exponent and CNN-LSTM', *The International Journal of Advanced Manufacturing Technology*, vol. 112, <https://doi.org/10.1007/s00170-020-06447-8>, pp. 2277-2299.

<https://doi.org/10.1007/s00170-020-06447-8>

DOI [10.1007/s00170-020-06447-8](https://doi.org/10.1007/s00170-020-06447-8)

ISSN 0268-3768

ESSN 1433-3015

Publisher: Springer

***The final publication is available at Springer via <http://dx.doi.org/10.1007/s00170-020-06447-8>***

Copyright © and Moral Rights are retained by the author(s) and/ or other copyright owners. A copy can be downloaded for personal non-commercial research or study, without prior permission or charge. This item cannot be reproduced or quoted extensively from without first obtaining permission in writing from the copyright holder(s). The content must not be changed in any way or sold commercially in any format or medium without the formal permission of the copyright holders.

This document is the author's post-print version, incorporating any revisions agreed during the peer-review process. Some differences between the published version and this version may remain and you are advised to consult the published version if you wish to cite from it.

# Prediction of the Remaining Useful Life of Cutting Tool using the Hurst Exponent and CNN-LSTM

Xiaoyang Zhang<sup>1</sup>, Xin Lu<sup>1</sup>, Weidong Li<sup>1,2\*</sup>, Sheng Wang<sup>1</sup>

<sup>1</sup> Faculty of Engineering, Environment and Computing, Coventry University, UK

<sup>2</sup> School of Logistics Engineering, Wuhan University of Technology, China

\* Corresponding author: weidong.li@coventry.ac.uk

## Abstract

To enhance production quality, productivity and energy consumption, it is paramount to predict the Remaining Useful Life (RUL) of a cutting tool accurately and efficiently. Deep learning algorithm-driven approaches have been actively explored in the research field though there are still potential areas to further enhance the performance of the approaches. In this research, to improve accuracy and expedite computational efficiency for predicting the RUL of cutting tools, a novel systematic methodology is designed to integrate strategies of signal partition and deep learning for effectively processing and analysing multi-sourced sensor signals collected throughout the lifecycle of a cutting tool. In more detail, the methodology consists of two subsystems: (i) a Hurst exponent-based method is developed to effectively partition complex and multi-sourced signals along the tool wear evolution; (ii) a hybrid CNN-LSTM algorithm is designed to combine feature extraction, fusion and regression in a systematic means to facilitate the prediction based on segmented signals. The system was validated using a case study with a large set of databases with multiple cutting tools and multi-sourced signals. Comprehensive comparisons between the proposed methodology and some other main-stream algorithms, such as CNN, LSTM, DNN and PCA, were carried out under the conditions of partitioned and un-partitioned signals. Benchmarks showed that, based on the case study in this research, the prediction accuracy of the proposed methodology reached 87.3%, which is significantly better than those of the comparative algorithms.

**Keywords:** Cutting tool life, Hurst exponent, CNN-LSTM, deep learning

## 1. Introduction

The condition of a cutting tool is considered as one of the most paramount factors to determine production quality, productivity and energy consumption (Yousefi and Zohoor 2019). If a computerised numerical control (CNC) machine is equipped with a system of tool condition monitoring and analysis for estimating the remaining useful life (RUL) of a cutting tool, it can prevent the insufficient utilisation of the tool or the poor surface quality/dimensional errors of workpieces due to the excessive wear of the tool. Investigations showed that 30% of the manufacturing cost could be saved by adopting such the system (Zhou and Xue 2018). Owing to the significance of the research, numerical works have been actively conducted in recent years.

The research of the RUL prediction on cutting tools can be classified into two general categories, i.e., an experience-based approach and a data-driven approach (Serin et al. 2020). An experience-based approach requires a large amount of machining cycles to be carried out to summarise laws of tool wear as mathematical models or some heuristic rules (e.g., the Taylor law). However, the status of a cutting tool is significantly influenced by various machining parameters, such as cutting speed and feed rate. Furthermore, as a cutting tool constantly contacts with a workpiece during machining, this process unavoidably involves many material-specific factors that stochastically affect the tool condition. Thus, it is difficult to generalise an experience-based approach for accurately predicting the RUL of a cutting tool in wider applications. Instead, a data-driven approach, which estimates the RUL of a cutting tool based on analysis of monitored signals from sensors mounted on a machine or a cutting tool, shows a greater potential for practical applications. Some earlier designs of such the approach depend on support vector machine (SVM) (Kaya et al. 2012), support vector regression (SVR) (Benkedjouh et al. 2015), artificial neural networks (ANN) (Shankar et al. 2018), fuzzy logic (Rajamani et al. 2018), etc., for analysing and estimating tool conditions and lives. However, those “conventional” intelligent algorithms still need tedious manual operations to define features to be extracted from monitored signals, and the pre-set features might not be consistent throughout an actual machining process. Another disadvantage is that those algorithms are incapable of handling a large volume of sensor data captured during the entire lifecycle of a machining process. In recent years, deep learning algorithms, such as the convolutional neural network (CNN), recurrent neural network (RNN) and long short-term memory network (LSTM), have been increasingly used to address the above challenges. The deep learning algorithms can facilitate a data-driven approach in extracting features from a large amount of monitored datasets more intelligently and autonomously. Nevertheless, industrial applications have been becoming more complex, requiring a more accurate and efficient approach. Therefore, there are some further exploration areas, including the followings:

- Each deep learning algorithm exhibits distinguishing characteristics but also shows some limits. For processing complex signals, there is an increasing trend to hybridise the architectures of well-performed deep learning algorithms to leverage their strengths. For the RUL prediction of cutting tools, signals could be from different sensor sources demonstrating various physical features. It is worth exploring how to design an appropriate hybrid deep learning algorithm to fuse signals and extract their inherent features effectively in supporting more accurate RUL estimation of cutting tools.
- Meanwhile, the complexities of signals for a cutting tool arise not only from different types of sensors but also from different stages of a cutting tool lifecycle. Features in the stages could be in dramatically unbalanced distributions. To further improve the performance of a deep learning algorithm in terms of accuracy and computational efficiency, it is imperative to develop an effective strategy to segment a large volume of signals wisely to facilitate the deep learning algorithm for signal processing.

This paper is aimed at addressing the above issues. The system proposed in this research consists of two subsystems: (i) to enhance the prediction accuracy and expedite computational efficiency, a Hurst exponent-based method is developed to partition complex signals along tool wear evolution; (ii) a hybrid CNN-LSTM algorithm, which combines feature extraction, fusion and regression, is designed and applied for predicting the RUL of cutting tools. The system is validated using a case study with a large set of signals from multiple sources. The innovative characteristics of the research are below:

- A novel systematic methodology is designed to integrate the strategies of signal partition and deep learning for effectively processing and analysing multi-sourced sensor signals throughout the lifecycle of a cutting tool;
- Detailed comparisons between the proposed system and other main-stream algorithms were conducted, and benchmarks were made to demonstrate the superior performance of the research.

The rest of this paper is organised as follows. The related works are reviewed in Section 2. The developed methodology is presented in Section 3. Section 4 presents methodology validation and benchmarking with comparative algorithms. Finally, conclusions are drawn in Section 5.

## **2. Related Works**

Flank wear, which occurs on the surface of a cutting tool that contacts with a workpiece, is the most common form of tool wear (Zhang et al. 2016). A lot of related works conducted analysis on flank wear as it is able to provide a more reliable RUL prediction on a cutting tool. To simplify the terminology, the flank wear of a cutting tool is called tool wear in the rest of the paper. As aforementioned, there are various dynamic and uncertain factors in practical machining processes, and an experiment-based approach is not effective in evaluating tool wear and estimating RUL effectively. Instead, a data-driven approach has been receiving more attentions considering the approach is more robust in supporting practical applications. The performance of the data-driven approach is highly dependent on the embedded intelligent algorithms and the mechanisms for signal processing.

Zhang et al. (2016) collected vibration signals in a machining process, and key features were identified based on the Pearson correlation coefficient (PCC). A neural-fuzzy network (NFN) was developed to predict tool wear and the corresponding RUL. The research provided comparisons between the NFN and an artificial neural network (ANN), and between the NFN and a radial basis function network (RBFN). Based on the results, it was concluded that the NFN had a better performance on the prediction of tool wear and RUL. Yu et al. (2017) developed a hidden Markov model (HMM)-based method for tool wear monitoring and RUL prediction. The root mean square (RMS) of vibration signals was calculated as a health indicator to construct a multiple HMM in a weighted manner for RUL prediction. In the method, tool wear was divided into discrete regions by the HMM to better envision the status and evolution of tool wear. Wu et al. (2018) proposed a multi-sensor fusion system for online prediction of RUL for cutting tools. In the method, first, the statistical features in the signals of vibration, acoustic emission and current in the time- and frequency- domains

were extracted. In considering the relationship between time-varying signals and cutting tool wear, features were manually segmented, and correlation analysis, monotonicity analysis and residual analysis were carried out for selecting the most relevant features to support the prediction. Then, an adaptive network-based fuzzy inference system (ANFIS) was applied to achieve feature fusion. Finally, the RUL prediction was conducted using a polynomial curve fitting algorithm. Yang et al. (2019) established a model for predicting tool RUL based on a trajectory similarity-based prediction (TSBP) algorithm and the differential evolution SVR (DE-SVR) algorithm. Signals of cutting force were collected during a machining process, and analysed in the time domain, frequency domain and wavelet to extract the most representative features. The root mean square value (RMSV), the average value (AV) and the standard deviation value (SDV) in the time- and frequency- domains were evaluated to investigate their correlation with tool wear. Then, these features were identified via their importance to be further imported into an integrated model for prediction.

In recent years, deep learning algorithms were leveraged to support more effective analysis and reasoning for the RUL prediction on cutting tools. For example, An et al. (2020) proposed an integrated model that combined a CNN and a stacked bi-directional and unidirectional LSTM (SBULSTM) to predict the RUL of a cutting tool. Among them, CNN was used to extract features from signals for dimensionality reduction, and then SBULSTM was used to train these features and achieve the purpose of prediction. More works can refer to recent reviews on the topic (Zhou and Xue 2018; Serin et al. 2020).

Although the reviewed works achieved some success in the tool RUL prediction, there are still obvious defects and potential improvement areas below:

To meet more complex industrial requirements and enhance the prediction accuracy, multiple types of sensors are increasingly adopted and a large amount of signals will be generated during processing. The approach based on a single machine learning algorithm could be hindered in processing high-dimensional inputs (Wu et al. 2019). With the growth of data, algorithms built could be no longer a single network architecture for data analysis. Recently, the study of more complex hybrid network models has been becoming a new trend (Zang et al. 2020). Such the state-of-the-art strategy integrates the strength of various learning algorithms and presents greater advantages to process the data sources that contain multiple characteristics. Among the hybrid models, a hybrid CNN-LSTM algorithm is promising in performance and was applied in some applications, such as action video recognition (Zhu et al. 2020), traffic forecasting (Bogaerts et al. 2020), residential energy consumption (Kim and Cho 2019), engineering (He et al. 2019). Based on the evidence of successful applications, it is valuable to consider designing an appropriate CNN-LSTM algorithm to perform the RUL prediction of cutting tools with a high performance.

Moreover, due to the imbalanced distributions of acquired signals during a processing, deep learning algorithms could lead to inaccurate prediction throughout the tool lifecycle. Therefore, it is useful to develop sensible strategies to segment sensor signals into fragments to promote the efficiency

and reliability of deep learning algorithms in processing the signals. Low et al. (2018) showed that the multi-model prediction of the RUL of a cutting tool based on several discrete partitions of data had a better performance than working on the entire data directly. Similarly, Opalka et al. (2018) demonstrated that a more accurate RUL prediction can be achieved by dividing time series sensor data into smaller segments according to a certain indicator. The Hurst exponent, which was developed by Hurst (1900-1978) based on the rescaled range (R/S) analysis method, has been extensively applied to determine the long-term memory of time series data that are varying in degree and time span (Knight and Nunes 2018). There are increasing uses of the Hurst exponents in signal segmentation to support different applications (Guan et al. 2018; Mohanty et al. 2018). In this research, an effective strategy of the Hurst exponents is explored in establishing correlations between signals and tool wear development, and then a sensible partitioning method for the signals is developed.

### 3. Methodology and System Flow

The workflow of the system and the methodology for the RUL prediction on cutting tools are shown in Figure 1.

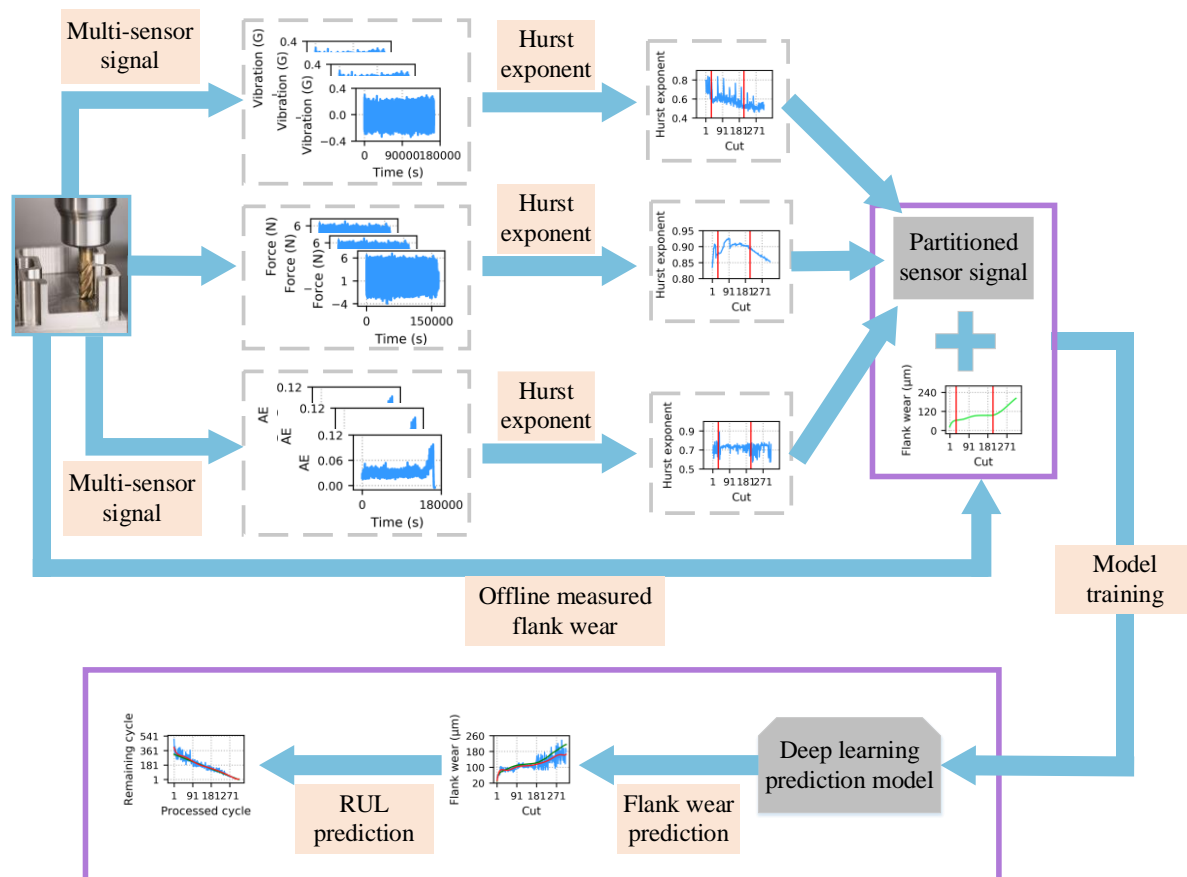


Figure 1. The flow and architecture of the presented system.

The system consists of two subsystems, i.e., the Hurst exponent-based data partition, and a hybrid CNN-LSTM algorithm for RUL prediction. In the system, three types of sensors are deployed to

monitor the statuses of a cutting tool, i.e., vibration (V), cutting force (F) and acoustic emission (AE). The signal partition method based on the Hurst exponent is employed to segment sensor data into small batches to minimise the impact of feature imbalances from the input data. The partitioned datasets are then sent to the CNN-LSTM algorithm for processing, where the RUL prediction on a cutting tool is generated. In the CNN-LSTM algorithm, each individual CNN is designed as a channel to process one type of signals, and the extracted features from the signals are fused in a concatenated layer, and the synchronised signals are further sent to the LSTM as a regression layer for predicting tool wear and RUL.

### 3.1 Signal partition based on the Hurst exponent

The input signals for the developed system are from multiple sensors. Multiple sensors V, F, and AE are mounted on a CNC machine. After a total  $k$  cuts are executed, each sensor collected  $N$  samples of data in each cut. The raw datasets  $\{M^i | i = 1, 2, \dots, k\}$  are organised to obtain the sub-datasets for each sensor. Then, the Hurst exponent is conducted for partition on the data of individual cut from each sensor to acquire the input for the CNN-LSTM algorithm.

Signals from different sensors could have different impacts on the RUL prediction. The Hurst exponent is employed to process various sensor signals in order to establish accurate correlations between the signals and the prediction of tool wear. Key parameters of the Hurst exponent for sensor signals can be obtained using the following steps (Borys 2020):

For each window size  $n$  between  $N$ ,  $N/2$ ,  $N/4$ ,  $N/8$ ...until  $n$  approaching 350, repeat Step 1 – Step 6:

Step 1: Divide a given time series of the sensor signal for each cut into  $M$  subseries of length  $n$ .

Then calculate the mean value ( $\bar{T}_m$ ) of the  $m^{th}$  subseries as follows:

$$\bar{T}_m = \frac{1}{n} \sum_{i=1}^n T_{i,m} \quad (1)$$

where,  $T_{i,m}$  stands for the  $i^{th}$  signal in the  $m^{th}$  subseries and  $m = 1, 2, \dots, M$

Step 2: Create the mean adjusted series  $D_m$  of the sensor signal  $T_{i,m}$ :

$$D_{i,m} = T_{i,m} - \bar{T}_m \quad \text{for } i = 1, 2 \dots n \quad (2)$$

Step 3: Calculate the cumulative deviate series  $Z_m$ :

$$Z_{j,m} = \sum_{i=1}^j D_{i,m} \quad \text{for } i = 1, 2 \dots n \quad (3)$$

Step 4: Calculate the range of  $R_m$ :

$$R_m = \max(Z_{1,m}, Z_{2,m}, \dots, Z_{n,m}) - \min(Z_{1,m}, Z_{2,m}, \dots, Z_{n,m}) \quad (4)$$

Step 5: Calculate the standard deviation,  $S_m$ , by using the following:

$$S_m = \sqrt{\frac{1}{n} \sum_{i=1}^n (T_{i,m} - \bar{T}_m)^2} \quad (5)$$

Step 6: Calculate the rescaled range  $R_m/S_m$  and average over the all the  $M$  samples:

$$R/S = \frac{1}{M} \sum_{m=1}^M R_m/S_m \quad (6)$$

Step 7: Finally, the Hurst exponent of the sensor signal of each cut is obtained by linear fitting the log values of the rescaled ranges  $R/S$  for all the window sizes  $n$  based on:

$$\log\left(\frac{R}{S}\right) = \log C + H \cdot \log n \quad (7)$$

where  $C$  is a constant,  $H$  is the Hurst exponent.

The value of a Hurst exponent (i.e.,  $H$  value) varies between 0 and 1, and it indicates the dependence of sensor signals on their past values (Lotfalinezhad and Maleki 2020). Based on the previous research, the following observations are made for various signals (e.g., V. F and AE in this research):

- a) When  $0 \leq H < 0.5$ , there is a negative correlation between the sensor signals and flank wear. A smaller  $H$  value implies that the signal fluctuation in this time period changes more dramatically.
- b) When  $H = 0.5$ , the corresponding signal presents the Brownian motion. It refers that the signal has no impact on future signals, and signal fluctuation will be completely random and unpredictable.
- c) When  $0.5 < H < 1$ , the fluctuation of the signal shows a continuous positive correlation between the sensor signals and flank wear. The future trend will follow the changes of the present signal, and this type of serial signal is predictable.
- d) When  $H = 1$ , time series signals will be in a straight line and there is no fluctuation and correlation.

Within the lifecycle of a cutting tool, the value of a Hurst exponent is obtained from each signal by using the above steps. The different values of the Hurst exponents will be used to establish the correlation between signals and various flank wear stages of a tool lifespan, which are normally divided into an initial region of rapid wear, a steady region of uniform wear and a severe region of dramatically increased wear. Accordingly, the sensor signals can be segmented based on the stages to facilitate the following prediction using a deep learning algorithm.

### 3.2 A hybrid CNN-LSTM algorithm for prediction

CNN has demonstrated a good performance at intelligently acquiring features from signals and is immune to the frequency variation in the signals. LSTM is more powerful for processing time series signals. However, both the algorithms present their own limitations when dealing with real-time signals. CNN assumes that all inputs and outputs are independent of each other in processing. The inherent information between features are neglected, resulting in performance degradation when processing time series signals (Fawaz et al. 2019). LSTM can extract long-term dependencies of features in the signal sequence to improve recognition accuracy. However, the training time for LSTM is much longer than that of CNN as it needs to learn the nonlinearity relationships in the signal. To leverage the characteristics of both algorithms, in this work, a hybrid CNN-LSTM algorithm is designed for better prediction on tool life. It is capable to identify the spatial and temporal relations in signals. Based on the different types of signals, multiple CNN sub-models are constructed as pre-processors. The CNN sub-model parallelly extracts the features from each sensor node, and these



features are then fused at a concatenate layer. Finally, the concatenated features from signals are sent to the LSTM for prediction. The proposed CNN-LSTM model is illustrated in Figure 2.

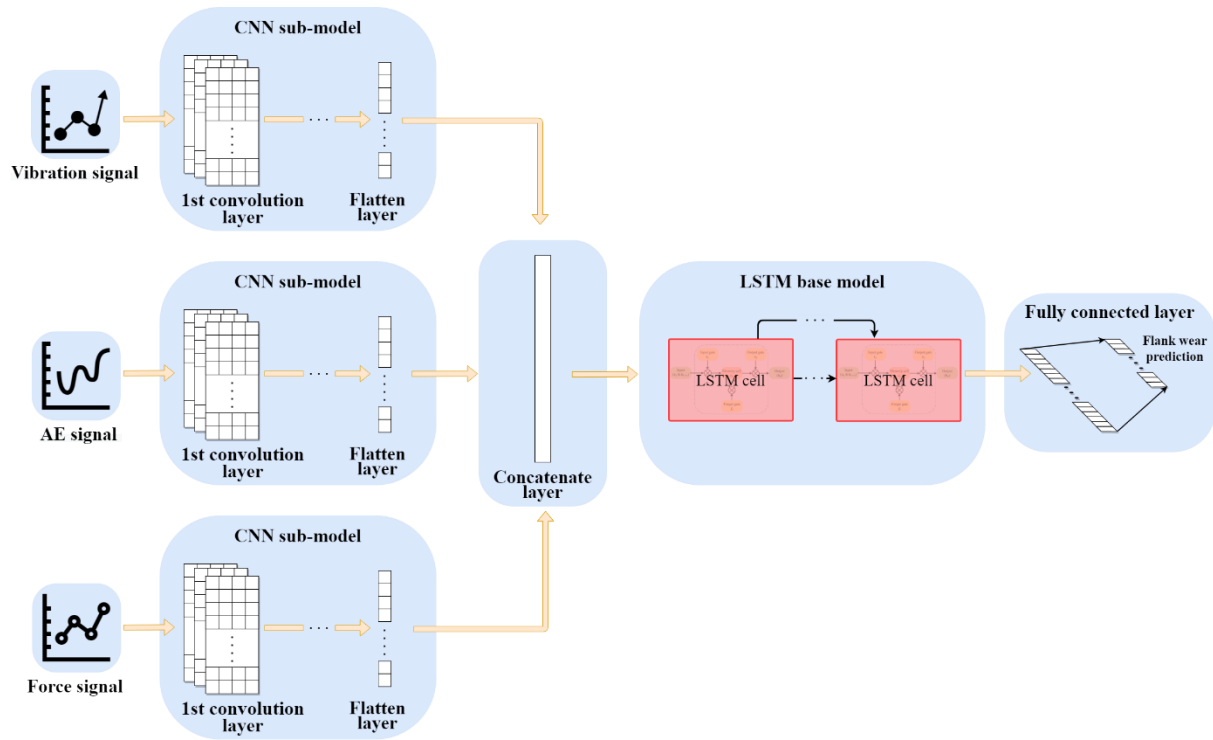


Figure 2. The architecture of the multiple channel hybrid CNN-LSTM model.

### 3.2.1 CNN design

CNN, which is one of the most successful deep learning models, offers the strength of extracting high-level dependency features automatically from input data. The learning ability and the training time of the CNN model are decided by its structure, especially the number of layers. Normally, a shallow structure cannot provide sufficient performance. Meanwhile, an excessive deep CNN may be harmful to the time sequential aspect of the data or cause overfitting (Chen et al. 2019, Zhu et al. 2020).

According to the different dimensionality of input variables, CNN is commonly classified into 1D and 2D structures. The 1D CNN is mainly used for sequential data processing, and 2D CNN is more suitable for image recognition (Yamashita et al. 2018). In the field of machinery condition diagnosis/prognosis, these two CNN architectures were employed on the basis of signals. For a 2D CNN, the format conversion of the time series data is necessary for matching the use of the 2D CNN. This was explored and discussed in the work of (Sun et al. 2017, Martínez-Arellano et al. 2019). Although, the conversion generates positive effects on the accuracy improvement of the prediction, the technology is complex and requires prior knowledge, so that it is perhaps the advised option only if there is an additional need (Fawaz et al. 2019). On the contrary, to process machinery time series signals with a 1D CNN has been proved more appropriate in several related studies, e.g. (Zhao et al. 2019, Li et al. 2020). Thus, a 1D CNN better utilises its characteristic of automatic feature extraction and avoids the deviation caused by manual signal processing (Chen et al. 2019). Moreover, based on

the prediction model architecture established in this work, the subsequent regression prediction of LSTM seeks time series signals as the input, rather than other formats, to preserve the temporal integrity of the signals.

Thanks to the great potential of a 1D CNN in processing sensor signals, a 1D CNN structure is adopted in this research. In order to achieve a satisfactory prediction accuracy and efficiency in processing sensor signals, the 1D CNN model with different numbers of convolution and pooling layers are evaluated to identify the most suitable configuration. The proposed model and several other models are summarised in Table 1. Based on the same input dataset (the exact dataset is described in Section 4), the prediction accuracies of the 1D CNN models are shown in Figure 3.

Table 1. The architecture of the adopted CNN model.

Model	Layer					
	1 <sup>st</sup> convolution	2 <sup>nd</sup> convolution	3 <sup>rd</sup> convolution	1 <sup>st</sup> pooling	4 <sup>th</sup> convolution	2 <sup>nd</sup> pooling
A	●	-	-	●	-	-
B	●	-	-	●	●	●
C	●	●	-	●	-	-
D	●	●	●	●	-	-
E	●	●	●	●	●	●
Proposed	●	●	-	●	●	●

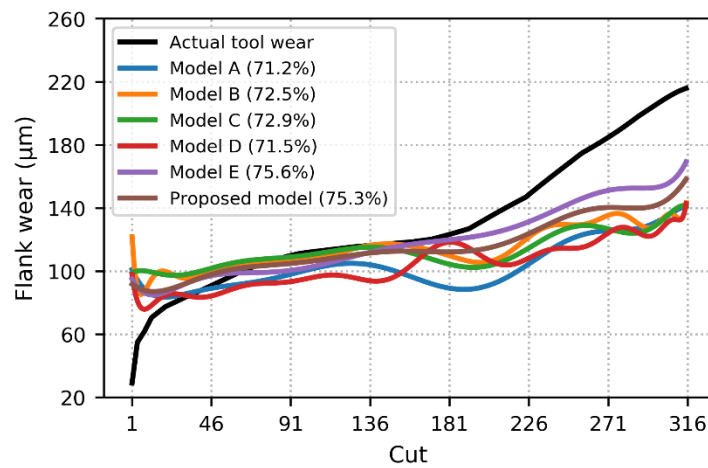


Figure 3. The prediction accuracies of the CNN models with different structure.

From the above results, it clearly shows that, within the six 1D CNN architectures, the best prediction accuracy was achieved by the model E, which is 75.6%. The accuracy of the proposed model of three convolutional layers and two pooling layers is 75.3%, which is slightly lower than that of the model E. However, the additional layer of the model E does not provide the significant improvement on the prediction accuracy. Meanwhile, considering manufacturing is becoming more customised, and cutting tools could be more diversified in terms of type during manufacturing. Thus, training of the algorithm on a new cutting tool is becoming more frequent. It is important to achieve a

balanced consideration of accuracy and algorithm training efficiency. Figure 4 displays the comparison of the memory usage of each model. Apparently, the model E with the six-layer structure needs more computing resources, and the computing load will increase exponentially in dealing with massive data. Thus, under comprehensive consideration, the proposed model provides the best trade-off in accuracy and efficiency performance among the six models.

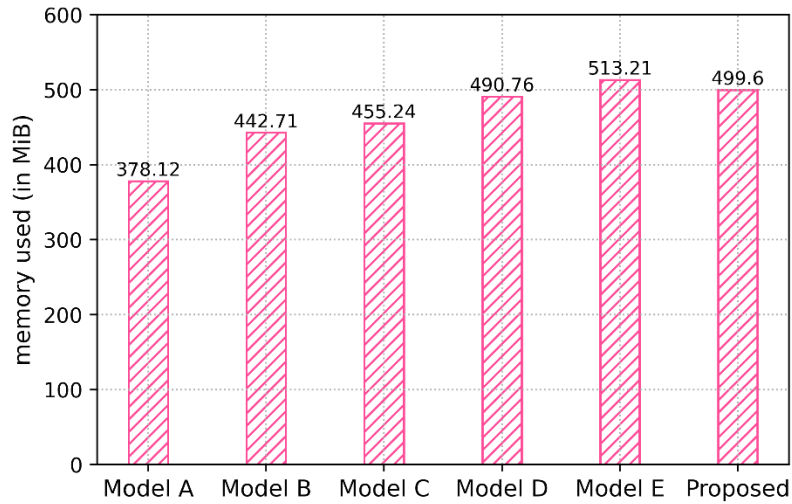


Figure 4. The memory usage of the CNN models with different structure.

Based on the selected 1D CNN architecture, pre-configured model parameters, such as the number of filters, kernel size, padding and stride, impact on the performance of the model. Among those parameters, stride determines the number of moving steps of the kernel in each convolution process. In order not to miss features in the original signals during the convolution, stride in this proposed model is set to 1. In addition, the model proposed in this work is responsible for processing a large number of sensor signals, and the convolution process helps to continuously reduce the amount of data, thereby improving processing efficiency. Thus, there is no need to apply padding, which ensures that the size of the matrix after convolution does not change by adding zero values to the input matrix. Furthermore, the kernel size determines the weight assignment in the convolution process. The smaller the kernel size, the process too extract features from the signals is more effective. However, a too small kernel size, such as 1x1 and 2x2, will lead to the concentration of local features, and lacks the connection between adjacent features. Thus, this work chooses 3x3 as the kernel size, which is also the most commonly used size in various applications. Finally, the number of filters usually depends on the input signals. An appropriate number of filters helps to achieve the optimal feature extraction and prediction accuracy, while an excessive number of filters will also increase the computational burden. In this research, the accuracy of four prevalently used number of filters, namely 16, 32, 64, 128, were compared, and the results is shown in Figure 5. Finally, the number of filters is chosen 64 as the accuracy is the highest.

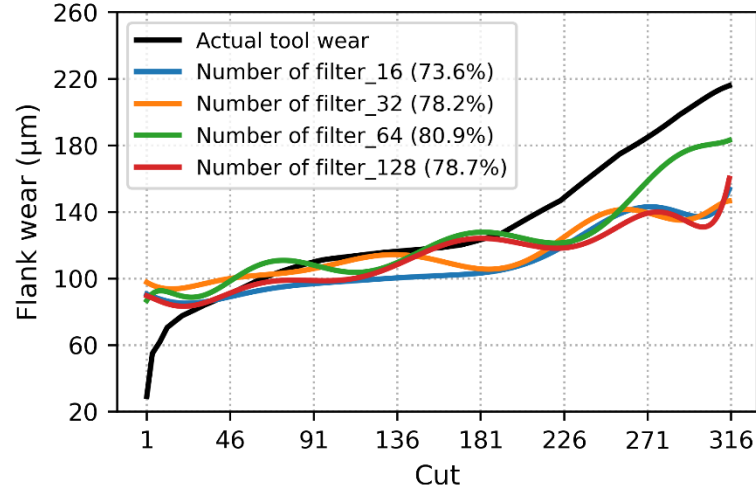


Figure 5. The prediction accuracies of the CNN model with different number of filters.

In view of the above analysis, the exact configurations of the proposed model are obtained as shown in Figure 6. To process the three types of sensor signals, i.e., V, F and AE, there are three such CNN models arranged in parallel (i.e., the CNN models of 1, 2 and 3 for the signals of V, F and AE respectively).

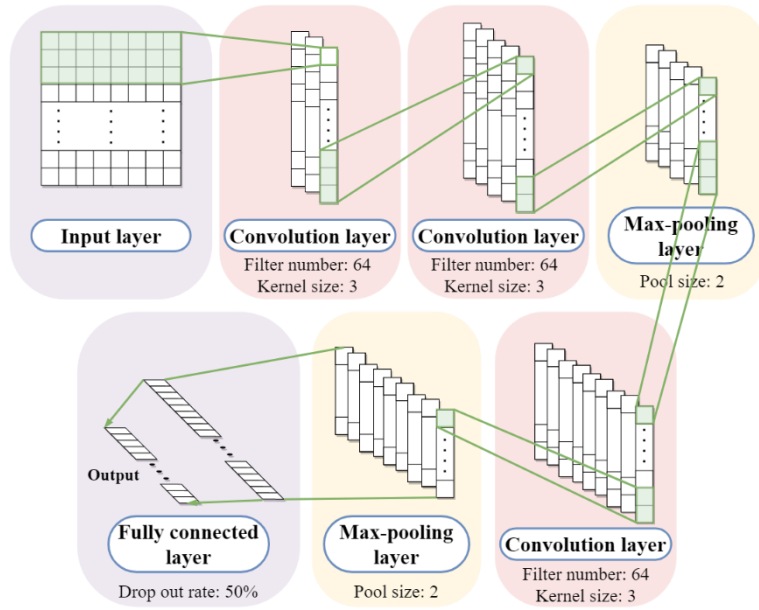


Figure 6. The proposed CNN architecture.

Each partitioned sensor signal in a format of matrix is fed into the CNN model through its input layer. For segmented sub-matrix for the inputs of the CNN models of 1, 2 and 3, they are below:

$$A = \begin{pmatrix} a_1^1 & a_2^1 & a_3^1 & \dots & a_n^1 \\ a_1^2 & a_2^2 & a_3^2 & \dots & a_n^2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_1^d & a_2^d & a_3^d & \dots & a_n^d \end{pmatrix} \quad B = \begin{pmatrix} b_1^1 & b_2^1 & b_3^1 & \dots & b_m^1 \\ b_1^2 & b_2^2 & b_3^2 & \dots & b_m^2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ b_1^d & b_2^d & b_3^d & \dots & b_m^d \end{pmatrix} \quad C = \begin{pmatrix} c_1^1 & c_2^1 & c_3^1 & \dots & c_e^1 \\ c_1^2 & c_2^2 & c_3^2 & \dots & c_e^2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ c_1^d & c_2^d & c_3^d & \dots & c_e^d \end{pmatrix} \quad (8)$$

where,  $n, m, e$  denotes the number of signals of sensor A, B and C, respectively,  $d$  denotes the number of cuts in the dataset after segmentation.

The convolution layer of each CNN model convolutes the input matrix to generate the spatial feature map by the activation function, and it can be described as:

$$\Phi_l = f(\text{conv}(X_{l-1} * w_l) + b_l) \quad (9)$$

where, the  $\Phi_l$  denotes the feature map of the  $l$ th convolution layer,  $X_{l-1}$  denotes the generated feature map from  $(l - 1)$ th layer,  $w_l$  denotes the weight,  $b_l$  denoted the bias, the  $\text{conv}(\cdot)$  denotes the convolution process. The  $f(\cdot)$  denotes the activation function. The rectified linear unit (ReLU) is selected as the activation function for each convolution layer in the proposed CNN model. The reason is that the ReLU increases the nonlinearity between layers, and only a small amount of computation is needed to alleviate the gradient vanishing problem (Carneiro et al. 2016).

After the convolution layer, the pooling layer is applied to further reduce the feature dimensionality, the max-pooling function is selected for every pooling layer, which extracts the maximum feature value with a window of size 2, so as to retain the important feature information and improve training efficiency. The output feature map can be depicted as:

$$\Phi_k = w_k \cdot \max(\Phi_l) + b_k \quad (10)$$

where, the  $\Phi_k$  denotes the feature map of  $k$ th pooling layer,  $w_k$  denotes the weight,  $b_k$  is the bias,  $\max(\cdot)$  denotes the max-pooling function.

Finally, the flatten layer is connected to the last pooling layer, to complete the transition from the convolution layer to the next layer base model by converting the generated features into a one-dimensional array. And the size of the output array equals the number of the cuts  $d$ .

So far, signal features extracted by the CNN model eliminate the interference between the heterogeneous sensors, and these features can be fused without complicated operations. Therefore, the concatenation layer is applied for the feature fusion after the multi-channel CNN models. As the element number of each feature vector is  $d$ , the feature fusion can be depicted as:

$$\begin{bmatrix} a'_1 \\ a'_2 \\ \vdots \\ a'_d \end{bmatrix} + \begin{bmatrix} b'_1 \\ b'_2 \\ \vdots \\ b'_d \end{bmatrix} + \begin{bmatrix} c'_1 \\ c'_2 \\ \vdots \\ c'_d \end{bmatrix} \rightarrow \begin{bmatrix} a'_1 & b'_1 & c'_1 \\ a'_2 & b'_2 & c'_2 \\ \vdots & \vdots & \vdots \\ a'_d & b'_d & c'_d \end{bmatrix} \quad (11)$$

where,  $a'$ ,  $b'$  and  $c'$  denotes the feature of different sensor generated from the CNN models.

### 3.2.2 LSTM

The focus of LSTM is to capture the dependencies of extracted features to achieve prediction. Once the data features of each CNN model have been merged at the concatenated layer, the generated column-wise array will be the input of the LSTM model, which can be expressed as  $X = [x_{i,j}]$ , where

$x$  denotes the feature value,  $i = \{1, 2, \dots, d\}$ ,  $j$  denotes different signals. A multi-layer LSTM uses  $X$  as input data, and each layer contains a LSTM cell (Figure 7).

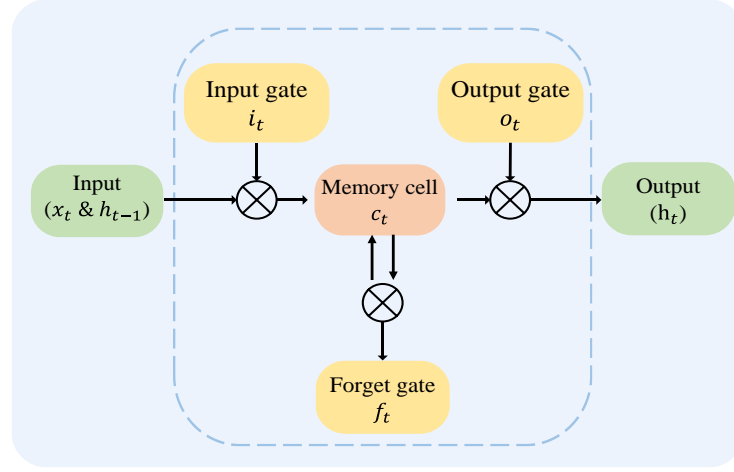


Figure 7. The LSTM cell in this research.

For the current time-step  $t$ , the output  $h_t$ , which indicates the predicted flank wear value, and the memory state  $c_t$  of LSTM cell are decided by the new input  $x_t$ , the output  $h_{t-1}$  and memory state  $c_{t-1}$  of the last time step  $t - 1$ . Based on the input array  $X$ , which contains  $d$  time-steps and  $j$  features in each time-step, the LSTM network will recursively obtain the comprehensive information from each time-step and provide corresponding predictions.

During the process, the forget gate  $f_t$  of the LSTM cell decides how many information of  $c_{t-1}$  should be forgotten according to the sigmoid activation function, and thus create the new feature data as candidate value. Then the candidate value is fed into the input gate  $i_t$  to update the memory cell state. Finally, the prediction value  $h_t$  of the LSTM cell at the time-step  $t$  can be calculated based on the updated cell state, which is from  $c_t$  and controlled by the output gate  $o_t$ . The above gate and cell can be obtained as follows:

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \quad (12)$$

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \quad (13)$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \quad (14)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tanh(W_c x_t + U_c h_{t-1} + b_c) \quad (15)$$

$$h_t = o_t \odot \tanh c_t \quad (16)$$

where,  $\sigma$  is the sigmoid activation function;  $\odot$  is the Hadamard product;  $W$  and  $U$  are variable weights and  $b$  is the bias.

Multiple LSTM cells are connected to form an LSTM network in time order, to predict the time sequence output (flank wear), that is  $\{h_1, h_2, \dots, h_d\}$ . And the number of the LSTM cell depends on the number of the feature in each signal received from the concatenation layer. The architecture of the LSTM model is shown in Figure 8.

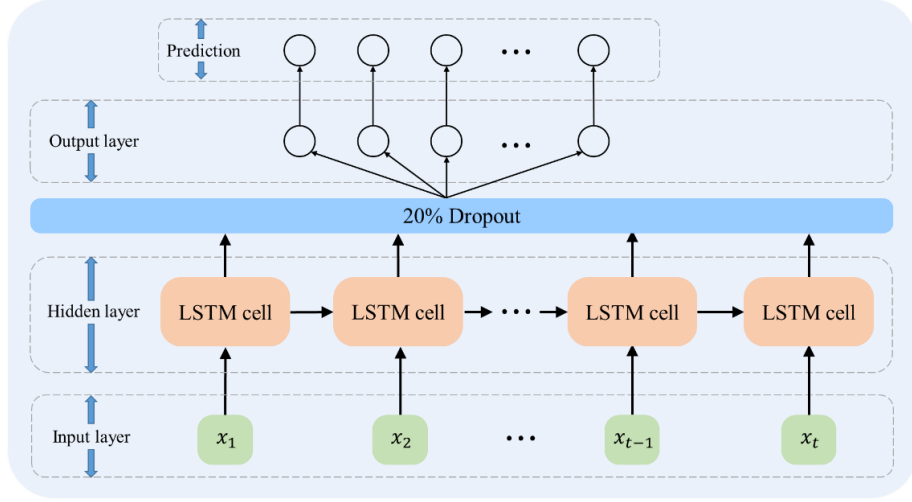


Figure 8. The LSTM architecture in this research.

Since the input data is partitioned via Hurst exponent into sub-datasets in this paper, the model training based on size-reduced sub-datasets may increase the possibility of overfitting. Therefore, inhibiting the overfitting of the proposed hybrid model is necessary to further improve the prediction performance. As the dropout is one of the most popular and efficient regularisation technologies used to prevent the overfitting, it has been employed in the LSTM model. Dropout randomly discards the hidden neuron with the setting dropout rate, and the remaining neuron is trained via backpropagation to obtain the new weight and bias (Liu et al. 2020). It can be described as:

$$S_i = \sum_j w_{ij} p_j S_j \quad (17)$$

where, the  $S_i$  denotes the output of  $i$ th layer after the dropout,  $S_j$  denotes the output of the  $j^{\text{th}}$  layer (previous layer),  $w$  denotes the weight,  $p$  denotes the dropout rate, which is set to 0.5 in this paper to maximise the regularisation.

Furthermore, in the last output layer of the hybrid model, the linear activation function is adopted to implement the regression prediction. For the prediction evaluation of the proposed hybrid CNN-LSTM model, the mean absolute error (MAE) is used as the criteria, which quantifies the absolute error between the prediction and actual values (Bhinge et al. 2017). It can be expressed as:

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |\tilde{y} - y| \quad (18)$$

where,  $n$  is the training sample size;  $\tilde{y}$  is the prediction value;  $y$  is the actual value.

## 4. Case Study and Methodology Validation

### 4.1 Experimental setup

To validate the methodologies, the dataset from the 2010 PHM Society Conference Data Challenge was adopted here (PHMSociety 2010). The dataset includes signals from cutting force sensors,

vibration sensors and acoustic emission sensors that were collected in the process of a dry milling on a high-speed CNC machine Roders Tech RFM760. In the experiment, a stainless-steel workpiece was machined using six 3-flute ball nose tungsten carbide cutters, and the corresponding sensor signals for the six cutting tools C1, C2, C3, C4, C5 and C6 were recorded. Every cutting tool was cut from new until significant wear, and a total of 315 cutting cycles were performed for each tool under the same machining parameters. In addition, three types of sensors are mounted on the workpiece and the machining table respectively, in terms of dynamometers, accelerometers and an acoustic emission sensor. The schematic of the experiment is shown in Figure 9.

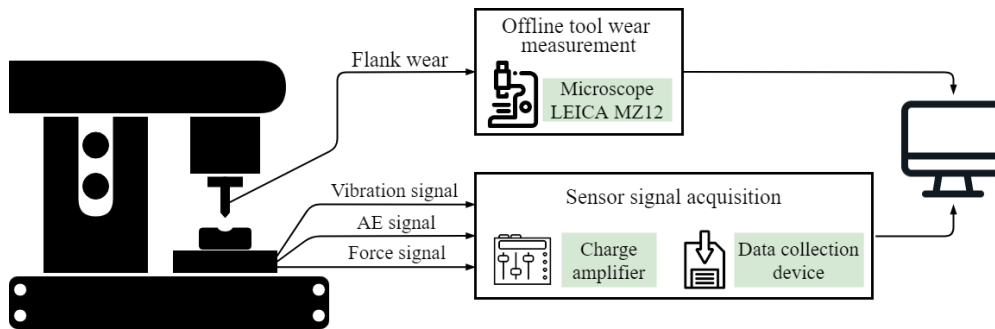


Figure 9. The schematic of the experiment.

During the machining process, three Kistler quartz 3-component platform dynamometers, three Kistler piezo accelerometers and a Kistler acoustic emission (AE) were used. These seven signal channels are shown in Table 2. A cutting tool processed the workpiece surface line-by-line along the x-axis with the axial depth of 0.2 mm, radial depth of 0.125 mm and cutting length of 108 mm per cut until the entire surface was removed. The spindle speed was maintained at 10,400 RPM and the feed rate was 1,555 mm/min. Moreover, after each cut, the cutting tools C1, C4 and C6 were placed under a LEICA MZ12 microscope to measure the flank wear ( $V_b$ ) of each flute, and three datasets that combined with sensor signals and flank wear of cutting tool C1, C4 and C6 were employed in this work. The dataset size of a single cutting tool is approximately 3.2 GB.

Table 2. Sensor signal of the dataset.

Sensor type	Signal sources
Kistler quartz dynamometer	Force (N) in the X axis, Force (N) in the Y axis, Force (N) in the Z axis
Kistler piezo accelerometers	Vibration (g) in the X axis, Vibration (g) in the Y axis, Vibration (g) in the Z axis
Kistler acoustic emission	AE-RMS (V)

## 4.2 Signal partition based on the Hurst exponent

Based on the datasets of the acquired sensor signal, the validation of the proposed system is implemented on a 3.60 GHz Intel (R) Core (TM) i7-7700 CPU processor (with 8.00 GB of RAM), in



which, the data processing and the computation of the deep learning algorithm were executed on the Keras framework and Tensorflow, respectively.

As aforementioned, 315 cuts were performed using the cutting tools C1, C4 and C6, respectively. Taking C4 as an example for analysis, the dynamometer signals on the X axis, vibration signal on the X axis and the AE sensor signals for the 1<sup>st</sup> cut and the last (315<sup>th</sup>) cut are plotted in Figure 10. It can be observed that the amplitude of the three sensor signals increases along with the machining process, indicating that the signals exhibit a good correlation with tool wear deterioration. Thus, these sensor signals are feasible to establish the prediction.

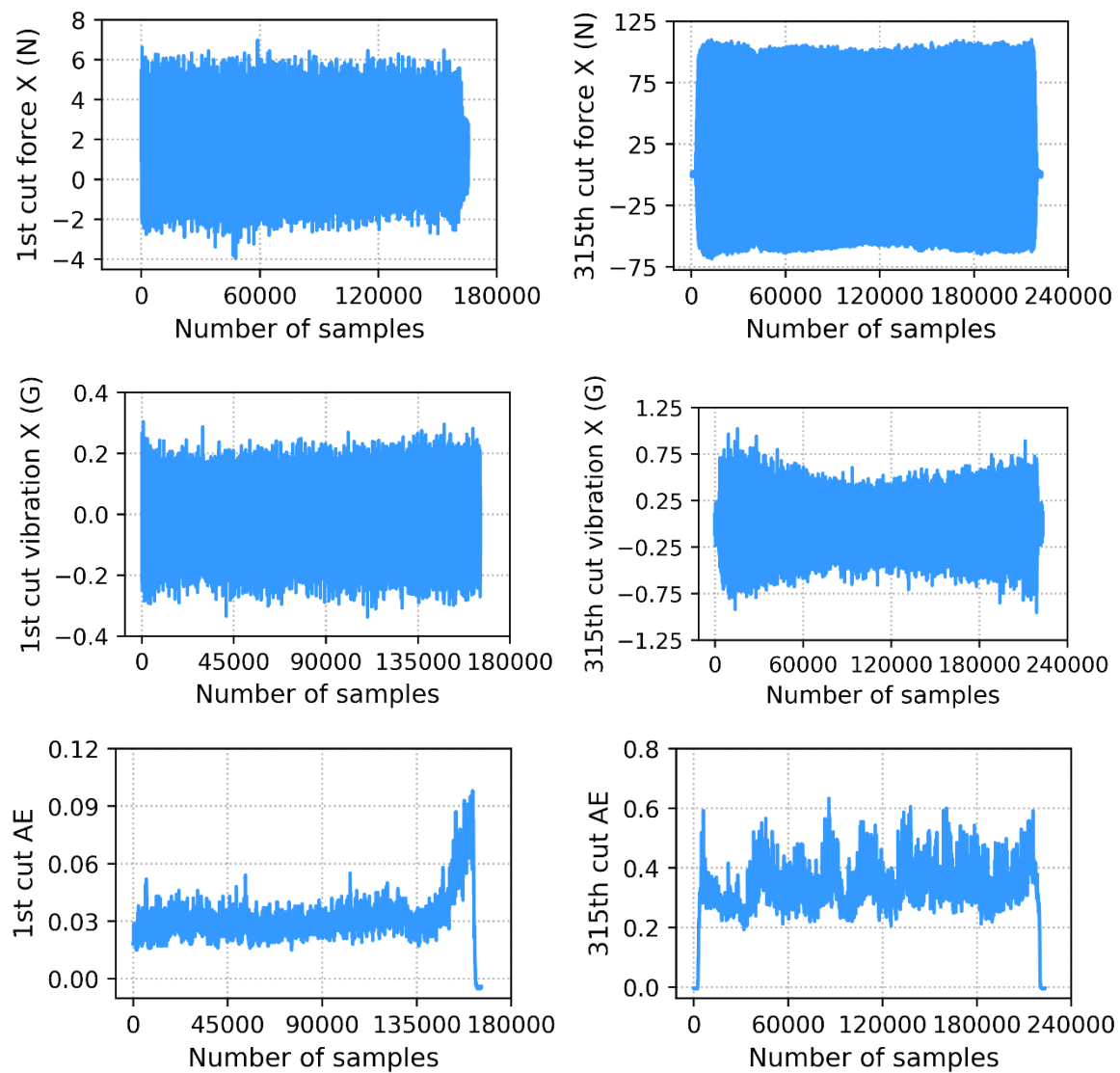


Figure 10. Sensor signals for the 1st and 315th cut by the cutting tool C4.

Corresponding to each cut, the flank wear of three flutes on the cutting tool was measured in the experiment. The flank wears of the cutting tool C4 are shown in Figure 11. According to the recommendation of the ISO 8688-2 (1989), the cutting tool life criterion is commonly predetermined by the average wear value of all flutes. Therefore, the average flank wears of the cutting tools C1, C4 and C6 were used for training the prediction algorithm presented in this paper.

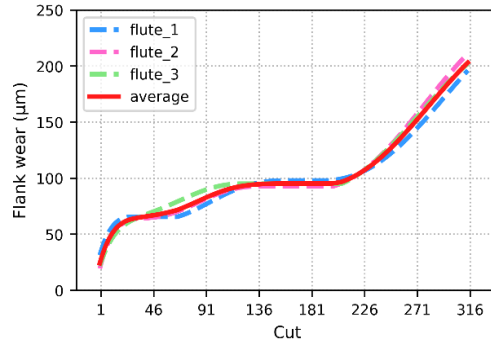


Figure 11. The flank wear of the cutting tool C4.

Tool wear involves different stages. Sensor signals over each stage is usually uneven, which may cause the wear prediction inaccurate. Moreover, as discussed earlier, due to the changing features of tool wear in the different stages, it is difficult for the prediction algorithm to effectively estimate different wear trends of tool wear during these stages. In this research, the Hurst exponent is used as an index to judge the fluctuations of sensor signals, and then the signals are segmented to correspond to the stages of flank wear. For the cutting tool C4, the Hurst exponent of three types of sensor signals are calculated for the 315 cuts. The Hurst exponents of the vibration signals in the X-axis, the cutting force signals in the X-axis and the AE signal are shown in Figure 12. Furthermore, to better represent the convergence and visualisation effect of the results, a cubic curve, which is a regression analysis method that preserves data characteristics and reduces data turbulence without changing the data, was applied to fit the Hurst exponent.

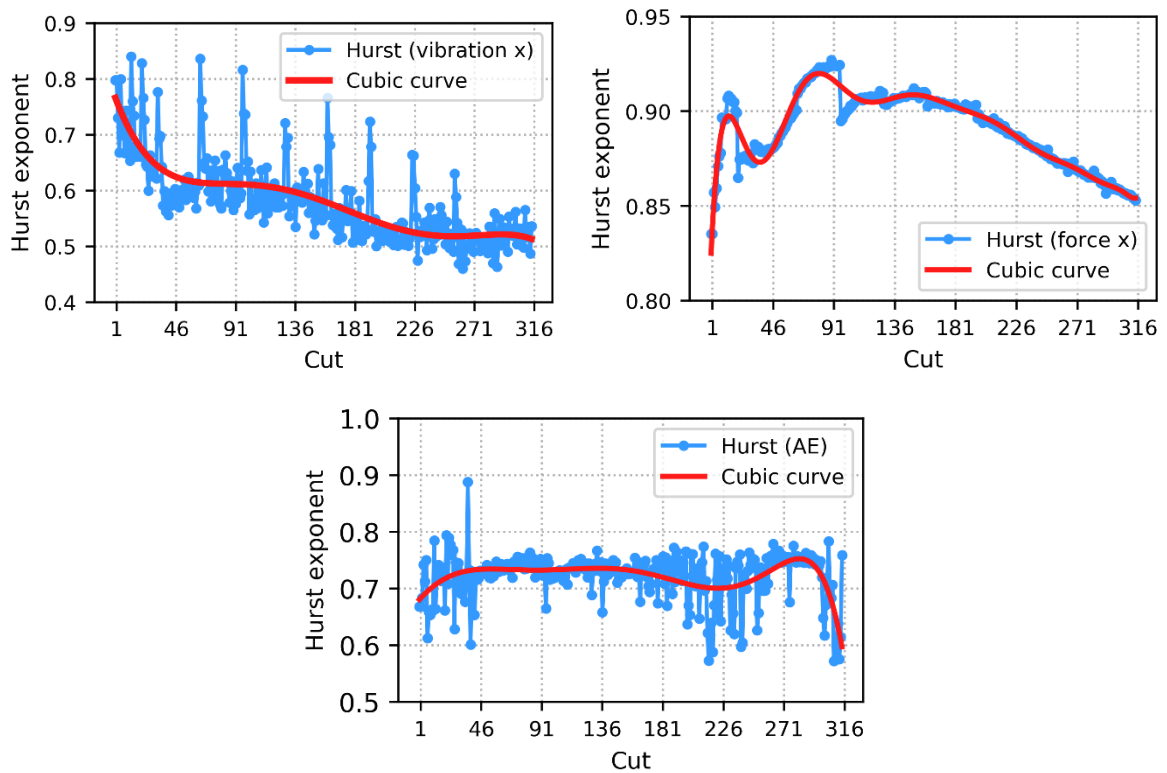


Figure 12. The Hurst exponent of the sensor signals (V, F and AE) for the cutting tool C4.

From Figure 12, the Hurst exponents (the  $H$  values) of V, F and AE are all roughly between 0.5 and 1. It means that these sensor signals present the persistent behaviours along with the tool wear and it is feasible to conduct prediction.

For the vibration signals, some observations are below:

- The  $H$  value of the cutting tool is the biggest at the beginning of machining, which is close to 0.85, implying that the signal has obvious regularity;
- As the machining progresses, the  $H$  values are higher than 0.6 before the 20<sup>th</sup> cut, so that the correlation between the vibration signal and the tool wear is still strong at this time. From the 20<sup>th</sup> cut to the 205<sup>th</sup> cut, the  $H$  values decrease to between 0.5 and 0.6, which indicates that the long-term memory of the signal is lower than its previous tool wear stage;
- After the 205<sup>th</sup> cut, the  $H$  values progressively approach to or even are lower than 0.5, meaning that the signals exhibit a trend of the Brownian motion. That is, the probability of negative correlation between the signal and the tool wear is increased;
- After the 315<sup>th</sup> cut, the  $H$  values are bound to drop below 0.5, so that the signal will completely show a negative correlation, and the cutting tool exceeds its health lifespan and the tool wear displays unpredictability;
- Overall, the fractal of the  $H$  values trend of the vibration signal has a strong correspondence with the different stages of tool wear, and the change of the  $H$  values are more sensitive. Thus, the vibration signal displays the greatest potential among the three signals for signal partition.

For the signals of the cutting force and AE, some observations are below:

- Along with the machining process until the 315<sup>th</sup> cut, the  $H$  values both show a decreased trend, which corresponds to the tool wear. Moreover, the  $H$  values are both greater than 0.5, it represents these sensor signals exhibit persistent behaviours and are positively correlated with the tool wear, thereby revealing that prediction of tool wear based on the cutting force and AE signals are also feasible;
- Despite this, it should be noted, the Hurst exponent trend of these signals are not as clear as the vibration signal in performing partition on the signals;
- This may be interpreted as: due to the acquisition frequency of the cutting force and AE signal is lower than that of the vibration signal, the collected noise signal that generated by excessive tool wear is insufficient. Moreover, the cutting temperature gradually increases with the progress of machining, so that it will soften the material and reduce the cutting force resistance (Xu et al. 2018). An AE sensor is prone to be affected by the mechanical noise from the background environment (Rusinek and Borowieca 2015);
- Therefore, the signals of three sensor are partitioned uniformly based on the Hurst exponents of the vibration signals in this work. All the signals were applied as the data sources for subsequent deep learning algorithm for the information compensation and purposes of prediction.

Accordingly, the above observations on signal changes are correlated with the stages of tool wear. For instance, for the cutting tool C4, it consists of the initial stage (1<sup>st</sup> – 19<sup>th</sup> cut), steady wear stage (20<sup>th</sup> – 204<sup>th</sup> cut), and severe wear stage (205<sup>th</sup> – 315<sup>th</sup>) as shown in Figure 13. The signal segmentations of three cutting tools are summarised in Table 3.

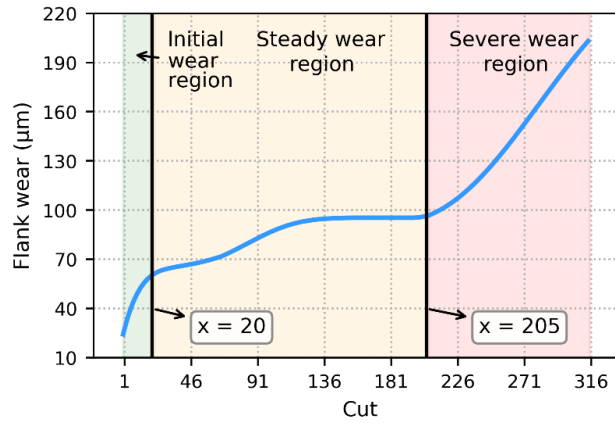


Figure 13. The stages of flank wear for the cutting tool C4.

Table 3. Signal segmentation of each cutting tool.

Cutting tool	Tool wear stages		
	Initial wear region	Steady wear region	Severe wear region
C1	1 to 49	50 to 139	140 to 315
C4	1 to 19	20 to 204	205 to 315
C6	1 to 14	15 to 179	180 to 315

Through the Hurst exponent, the segmentations of sensor data that correspond to different tool wear stages are identified. In this paper, three cutting tools, i.e., C1, C4 and C6, were used. To further improve the accuracy of prediction, the segmented sensor data were pair-wisely combined as the input dataset. For example, the sensor signal and flank wear of C1 and C4 were combined as a training dataset (denoted as C1C4), and the sensor signal and flank wear of C6 was treated as the validation dataset at the same time (refer to Table 4).

#### 4.3 Performance evaluation on the Hurst exponent and the CNN-LSTM algorithm

To evaluate the performance of the Hurst exponent-based partition, un-segmented signals (raw signals of each sensor) were used to perform prediction on flank wear based on the designed CNN-LSTM algorithm. The prediction accuracy of each dataset in this work is obtained by its corresponding validation set. And the prediction curves for the sensor signals based on the CNN-LSTM algorithm are shown in Figure 14.

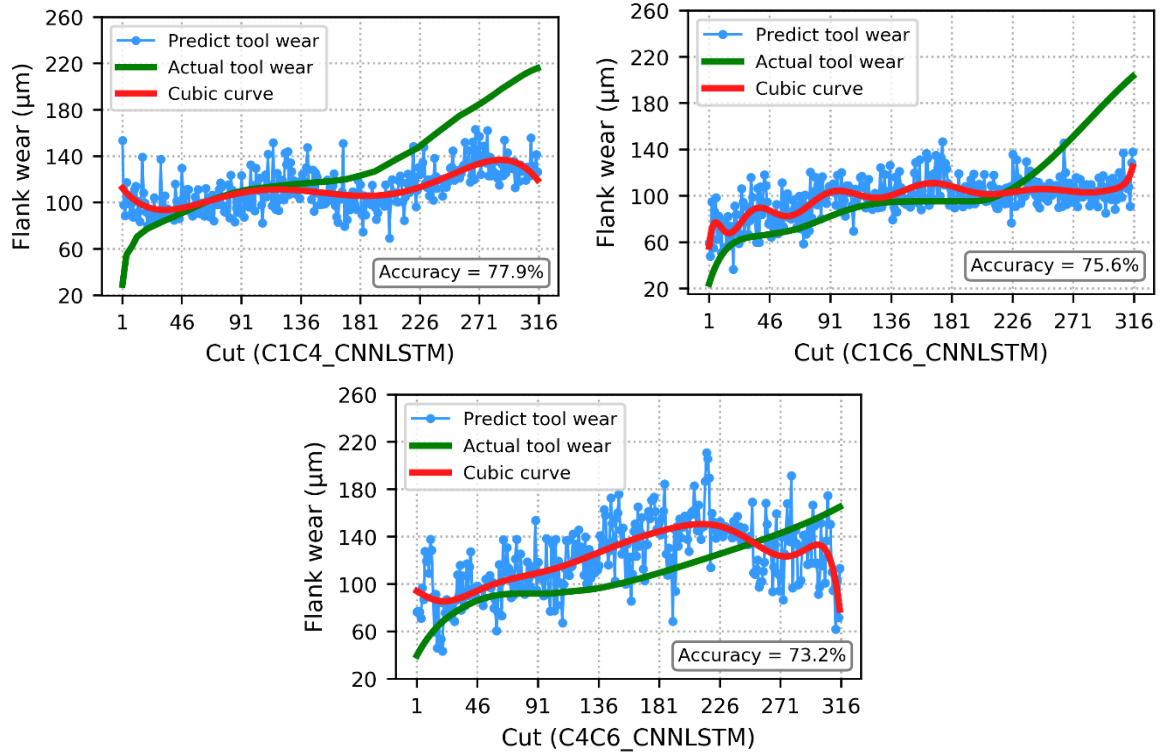
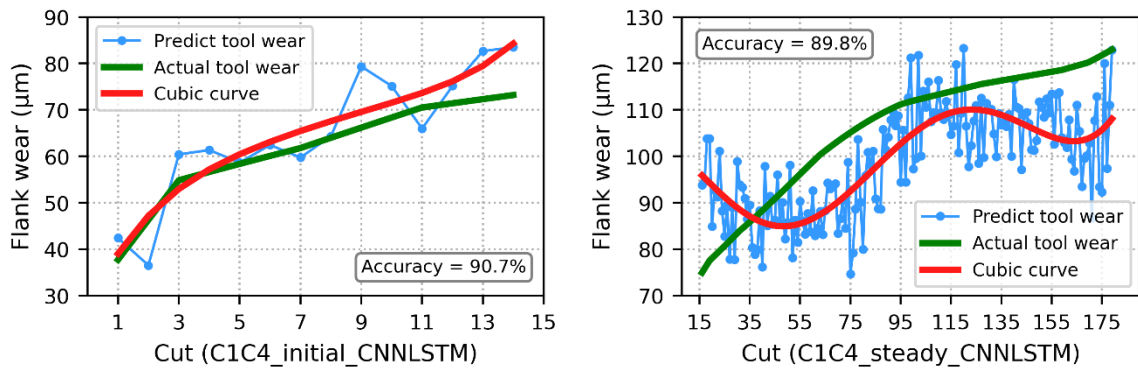


Figure 14. Prediction curves of the sensor signals based on the CNN-LSTM.

From the results, prediction accuracies are not satisfactory. It is expected to improve the prediction performance by adopting partitioned dataset according to the Hurst exponent. To do that, the sensor signals of the three cutting tools were divided into three groups according to the three stages of tool wear, and these signals were combined with corresponding flank wear values to form the new datasets. Figure 15 shows the prediction results for the signals of C1C4 based on the Hurst exponent.



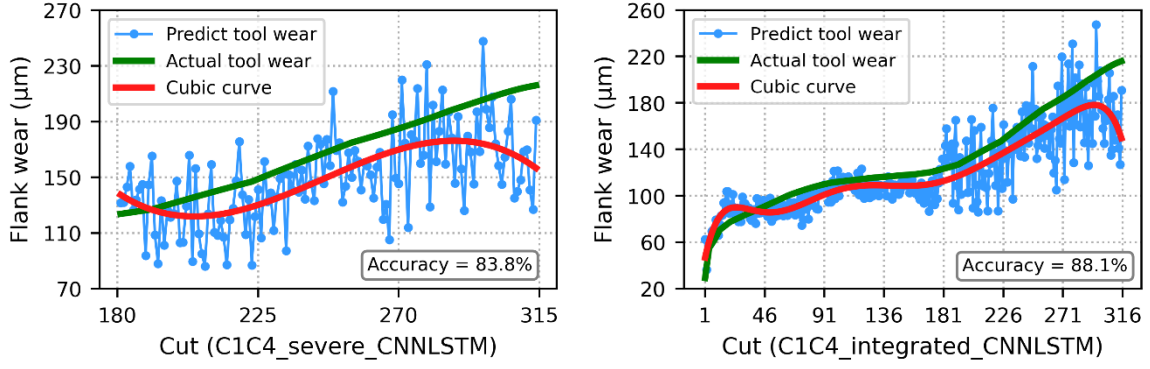


Figure 15. Prediction results based on the partitioned signals for C1C4.

For C1C4, the prediction accuracies in the initial wear stage, the steady and severe wear regions were 90.7%, 89.8% and 83.8%, respectively. The integrated prediction accuracy increased to 88.1% in comparison with 77.9% for the un-partitioned signals. The same experiments were conducted for the other datasets. As summarised in Table 4, for C1C4, C1C6, and C4C6, the accuracies using the Hurst exponent were improved by 10.1%, 10.4% and 14.5%, respectively.

Table 4. Comparisons of prediction accuracy for partitioned and un-partitioned datasets.

Dataset	Test	Prediction accuracy				
		Partitioned dataset				Un-partitioned dataset
		Initial stage	Steady stage	Severe stage	Integrated	
C1C4	C6	90.7%	89.8%	83.8%	88.1%	77.9%
C1C6	C4	87.4%	88.6%	87%	86.0%	75.6%
C4C6	C1	89%	87.8%	86.3%	87.7%	73.2%

Moreover, to further evaluate the performance of the developed research, the CNN-LSTM algorithm was compared with other hybrid deep learning architectures, which are CNN-CNN (Cheng et al. 2020), LSTM-LSTM (Choi and Lee 2018) and DNN-DNN (Zhang et al. 2019). To better compare the architectures, the structure and parameter of individual CNN and LSTM remained the same as the developed CNN-LSTM algorithm presented in Section 3. For the DNN model, it was designed with an input layer, an output layer and three hidden layers, in which the number of neurons in the input layer is equal to the sample number of sensor signals, each hidden layer is set as twice the number of input signal after multiple tests, and the size of the output layer depends on the number of flank wear values. The structure of the DNN is shown in Figure 16.

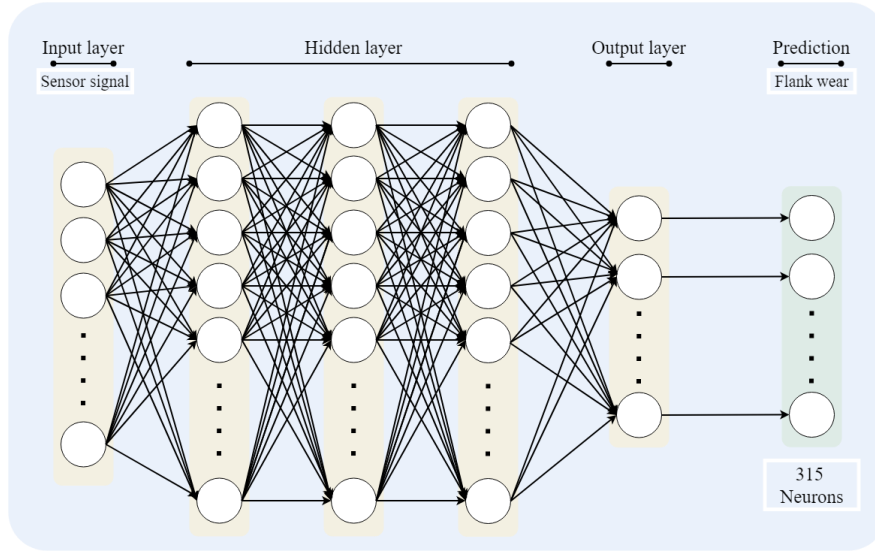


Figure 16. The DNN architecture.

The overall sensor signal dataset and the partitioned signal dataset were adopted to execute the prediction on the CNN-CNN, LSTM-LSTM and DNN-DNN respectively. Taking the dataset C1C4 as the example, the prediction results are presented in Figure 17 (un-partitioned signals) and Figure 18 (partitioned signals).

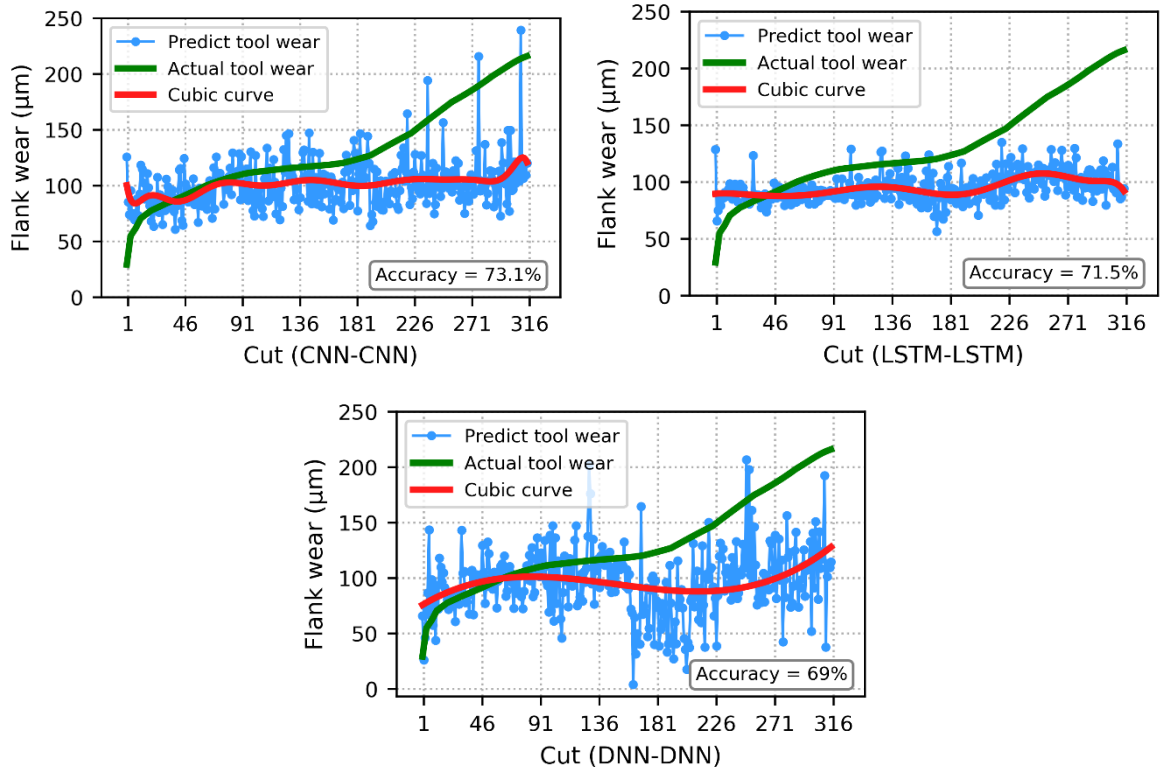


Figure 17. Prediction using the un-partitioned dataset C1C4.



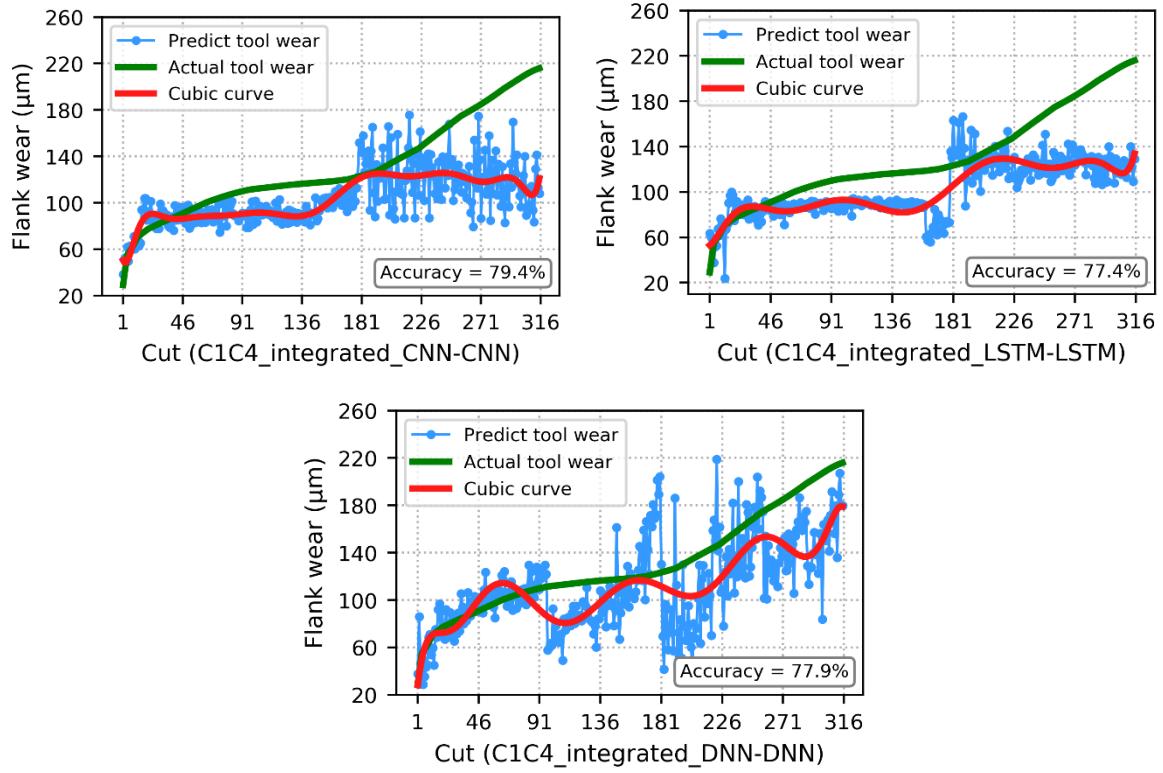


Figure 18. Prediction using the partitioned dataset C1C4.

The integrated result above of the dataset C1C4 is obtained after the prediction based on the corresponding sub-dataset of the tool wear initial, steady and severe stage. Figure 19 shows several representative fitting results of the individual sub-datasets, which covers different dataset, wear stage and deep learning model, and the prediction precision of the three ensemble models on the partitioned sub-datasets and the overall datasets is summarised in Table 5.

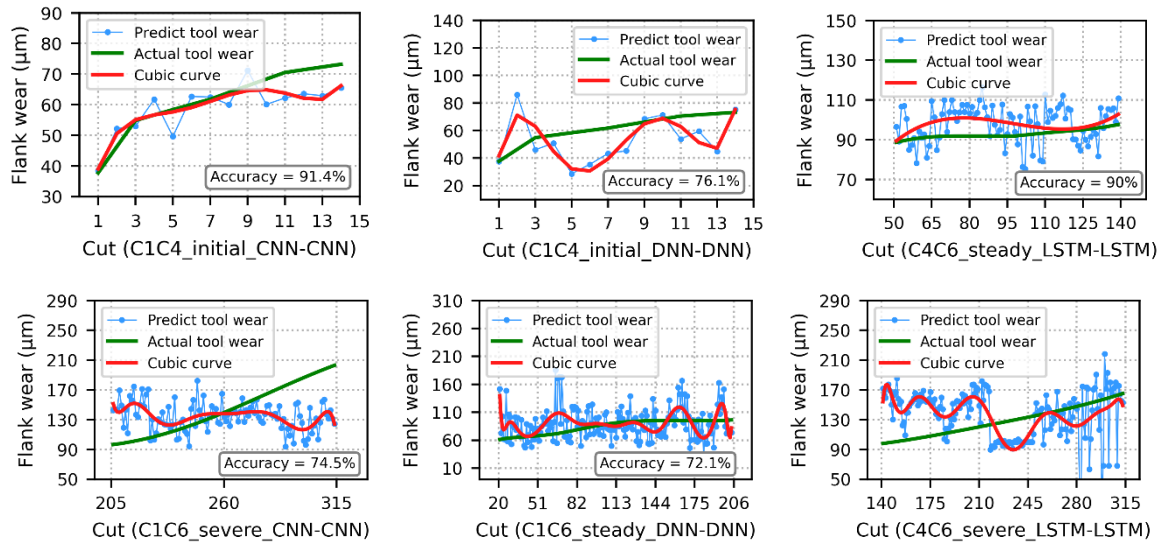


Figure 19. Curve fitting of the partitioned datasets on prediction models.



Table 5. Prediction precision of CNN, LSTM and DNN models.

Model	Dataset	Prediction accuracy				
		Partitioned signals				Un-partitioned signals
		Initial region	Steady region	Severe region	Integrated accuracy	
CNN-CNN	C1C4	91.4%	71.6%	81.2%	79.4%	73.1%
	C1C6	70.4%	79.5%	74.5%	76.9%	70.2%
	C4C6	81.1%	87.1%	67.7%	78.1%	70.8%
LSTM-LSTM	C1C4	80%	79.3%	74.3%	77.4%	71.5%
	C1C6	84%	78.4%	72.1%	75.5%	60.6%
	C4C6	80.7%	90%	71.4%	78%	65.4%
DNN-DNN	C1C4	76.1%	81.9%	73.4%	77.9%	69%
	C1C6	76.8%	72.1%	70.3%	71.6%	69.5%
	C4C6	72.7%	83.6%	70.7%	74.5%	63.9%

By comparing the results of Table 4 and Table 5, it is obvious that, even if the un-partitioned datasets were adopted as the input, the prediction accuracy of the CNN-LSTM algorithm was higher than those of other deep learning algorithms. That is, the average accuracy of the CNN-LSTM algorithm is 75.6%, while those of the CNN-CNN, LSTM-LSTM and DNN-DNN algorithms were 71.3%, 65.8% and 67.4%, respectively. It is expected, since the proposed hybrid model integrates the advantages of the automatic feature extraction of CNN and the time series data sequence learning of LSTM. Moreover, from the perspective of employing the partitioned dataset, the accuracies of the four algorithms were all improved to some extents. The CNN-LSTM algorithm achieved the best average performance of 87.3%, followed by the CNN-CNN algorithm of 78.1%. Such a result proves again that, the proposed data partition strategy of the Hurst exponent is effective, which the segmented sensor data is beneficial for optimising the prediction.

Computing efficiency is also a vital factor to investigate. In this research, the cumulative calculations of the computing time cost were carried out for the deep learning algorithms. The time usage of each prediction model to compute un-partitioned datasets and partitioned datasets are shown in Figure 20.

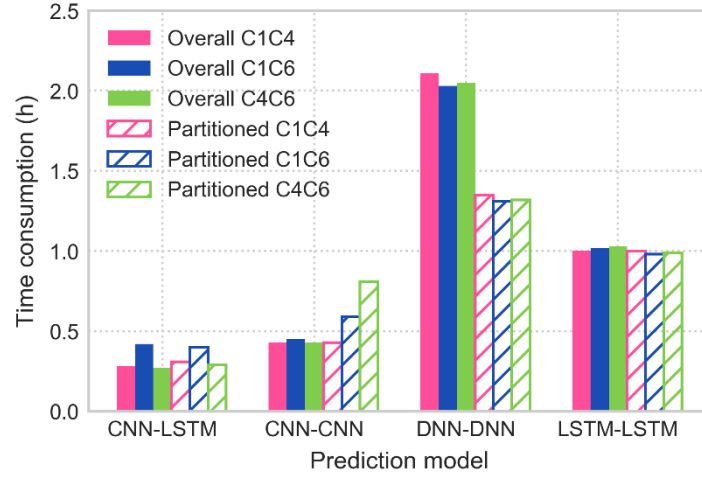


Figure 20. Time consumption of the prediction algorithms.

From the aspect of model training time consumption, it can be clearly seen that whether the un-partitioned or the partitioned data was used as the input for training, the proposed CNN-LSTM consumes the minimum computing time among all four algorithms. The average time for the CNN-LSTM algorithm was approximate 0.33h, the average time cost of the CNN-CNN model was the closest to the CNN-LSTM algorithm, about 0.52h. The DNN-DNN and LSTM-LSTM algorithms consume 80% and 67% more time than the CNN-LSTM algorithm, respectively. In view of the CNN-LSTM model being outstanding in the prediction precision and efficiency, it can be considered a powerful and promising deep learning scheme for the cutting tool RUL regression prediction.

#### 4.4 Comparison of the Hurst exponent with other methods

Furthermore, to effectively investigate the adaptability of the Hurst exponent-based signal partition method, the performance comparison between the proposed method and other prevalent signal processing approaches were executed. These approaches are domain-feature extraction and PCA dimensionality reduction.

The signal processing of the prediction task is responsible to eliminate unwanted data from massive sensor signals, to improve the efficiency and accuracy of the prediction model with lower volume and valuable data. For the domain-feature extraction, the features of the signals adopted in this paper are extracted in the time domain, frequency domain and time-frequency domain. In the time domain, statistical features are usually extracted to reflect the change of signal properties over time. And, as another indispensable feature extraction method, the frequency domain signal is able to present the signal change rate and provide the spectral feature for hidden information revealing (Herff and Krusienski 2018). The fast Fourier transform (FFT) is the most common frequency domain transformation. It is used to convert the sensor signals of each cut cycle in this paper. The FFT of the x-axis vibration signal of the 1<sup>st</sup> cut of the cutting tool C1 is shown in Figure 21. Moreover, facing the nonlinear sensor signal, the time-frequency domain signal outperforms the methods to process signals

in the time domain and frequency domain by capturing useful features, since it is conducive to explore the feature in transient and localized components. In this paper, the Continuous Wavelet transform (CWT) is applied to transform each sensor signal to the time-frequency domain, which is a powerful method to represent the sensor signal into a two-dimensional plane (Lee et al. 2017). The CWT result of vibration signals of the 1<sup>st</sup> cut for C1 in the X-axis is shown in Figure 22.

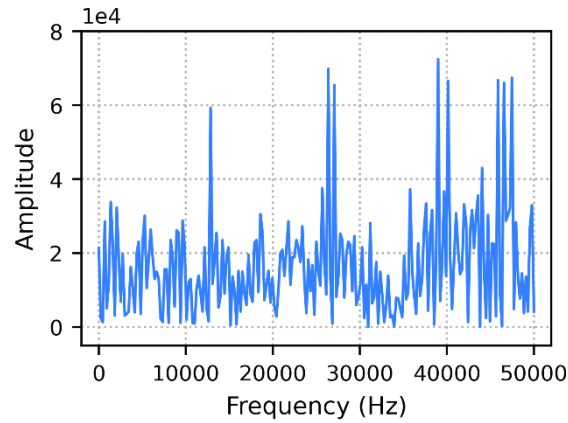


Figure 21. FFT spectrum of the x-axis vibration signal of 1<sup>st</sup> cut of the cutting tool C1.

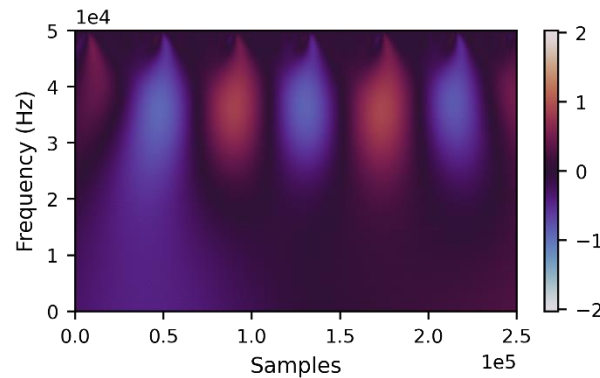


Figure 22. The CWT plot of the x-axis vibration signal of the 1<sup>st</sup> cutting for the cutting tool C1.

After the conversion of the frequency domain and time-frequency domain for sensor signals of the 315 cuts of each cutting tool, including the time domain, several widely employed features are extracted in every domain. It is noteworthy that the feature extraction in the time-frequency domain is based on a different scale, which is divided according to the sampling rate of the adopted dataset. Table 6 summarises the extracted features of three domains in this paper.

Table 6. Extracted feature from the different domain.

Domain			
Time domain	Frequency domain	Time-frequency domain	
Mean	Frequency centre	0-10000 Hz	Mean
Standard deviation	Median frequency	10000-20000 Hz	Standard deviation

Skewness	Root variance frequency	20000-30000 Hz	Variance
Kurtosis		30000-40000 Hz	
Maximum		40000-50000 Hz	
Peak to peak			

The validation dataset in this paper contains 7 sensor signals for the 315 cut cycles of 3 cutting tools. Finally, there are 158,760 features obtained from raw sensor signal by the above feature extraction mentioned. Then, the generated features of each cutting tool are gathered as a new input dataset, and they are adopted to perform the flank wear prediction on the CNN-LSTM algorithm, the fitting result is shown in Figure 23.

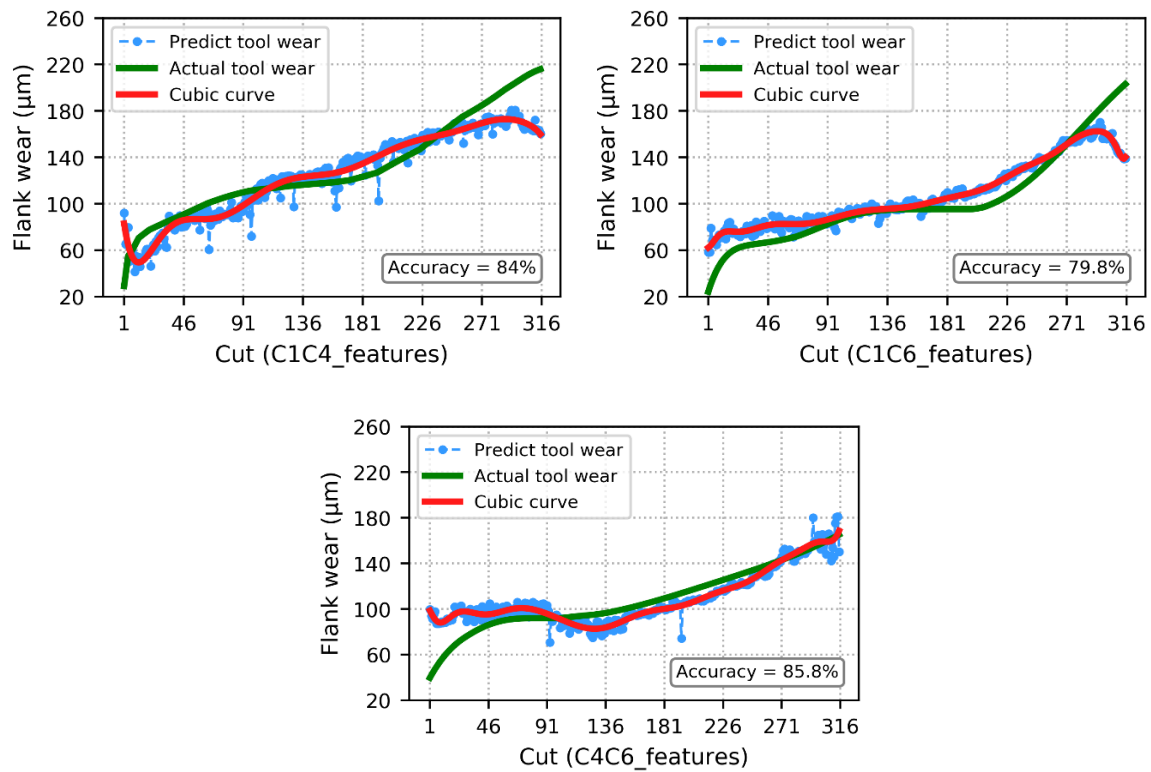


Figure 23. Fitting results of the domain feature input by the CNN-LSTM algorithm.

Besides the domain-feature extraction, the signal processing based on the PCA is a prevailing approach for dimensionality reduction of the multi-sensor signal. It is capable to compress the signal dimension without discarding main information by mapping the raw sensor signal into a feature space with representative components (Li et al. 2017). For the dimensionality reduction of the sensor signal by PCA, the number of components is necessary to be determined initially via the cumulative sum of explained variances, which provides the proportion of the retain information of the different number of components. Taking cutting tool C1 as an example, the cumulative sum of explained variances of the sensor signal in the 1<sup>st</sup> cut is shown in Figure 24.

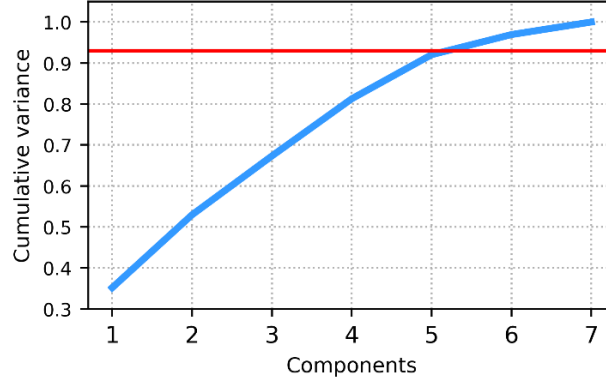


Figure 24. The PCA cumulative variance plot for the 1<sup>st</sup> cutting of the cutting tool C1.

From the above figure, it is observed that five components generated by the PCA could remain over 90% original sensor signal information, which is the same outcome of the other two cutting tools. Thus, these five components are adopted to replace the raw sensor signals of every cutting tool, each component includes the same number of data as the original sensor signal. Then, the flank wear prediction based on the five components was executed by the CNN-LSTM algorithm. And the input strategy is pair-wise concatenated as well. Figure 25 shows the prediction result.

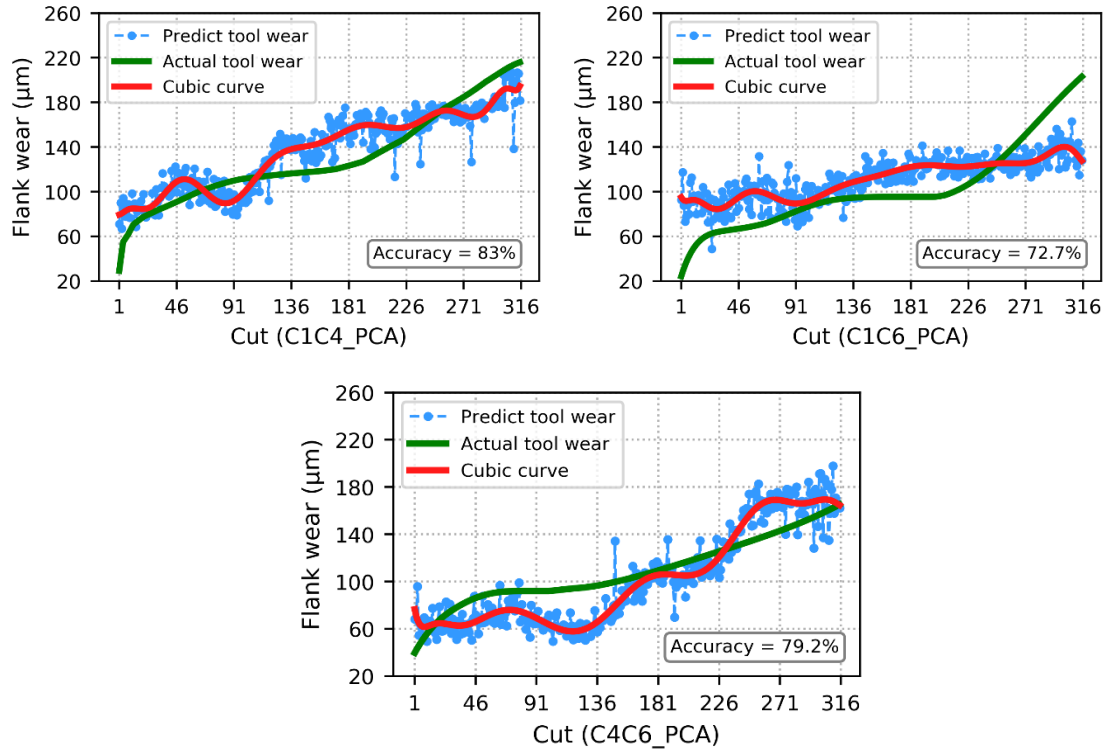


Figure 25. The fitting results of the PCA component input on the CNN-LSTM algorithm.

Based on the described approach of the domain feature extraction and the PCA dimensionality reduction, the corresponding flank wear prediction result on the hybrid CNN-LSTM model are detailed in Table 7. Figure 26 shows the total processing time consumption of the sensor signals of the three

datasets using these two methods and the proposed Hurst exponent method, as well as the cumulative time for their subsequent predictions.

Table 7. Prediction result of the different data processing methods.

Data processing method	Input dataset	Prediction accuracy	Average accuracy
Domain feature extraction	C1C4	84%	83.2%
	C1C6	79.8%	
	C4C6	85.8%	
PCA dimensionality reduction	C1C4	83%	78.3%
	C1C6	72.7%	
	C4C6	79.2%	

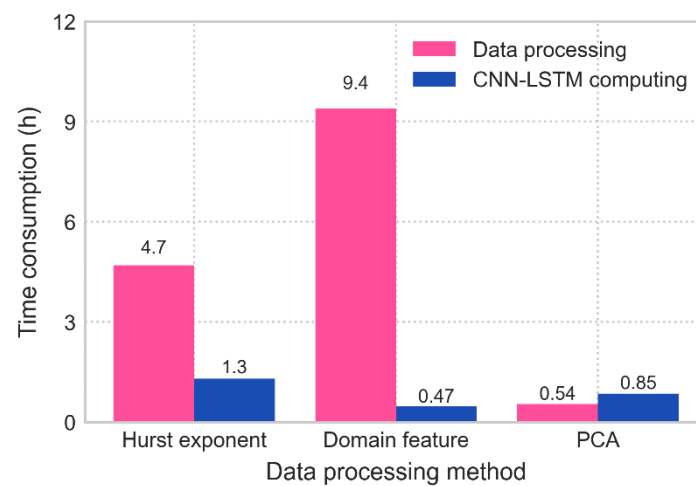


Figure 26. Time consumption of different data processing methods

By comparing the result of Table 4 and Table 7, it is apparent that the prediction accuracy for tool wear based on the dataset partitioned through the proposed Hurst exponent was higher than applying the domain feature extraction and the PCA dimensionality reduction. Separately, the feature extraction in the three domains achieved the data volume decreasing to a great extent. Due to the elimination of redundant signals, its precision was closer to the Hurst exponent-based partition method and the model calculation time was also compressed greatly. However, the domain conversion and feature extraction consumed a large amount of time, which were 50% and 94% higher than those of the Hurst exponent method and the PCA dimensionality reduction, respectively. It is an obvious obstacle to practical applications. On the other hand, PCA showed the benefit of the computational time-cost saving, by condensing the raw sensor signal into five new representative components, an approximate 30% dimensionality reduction was implemented. Nevertheless, the prediction accuracy is sacrificed to reach this dimensionality reduction, and it is foreseen that the optimal components of the PCA will appear more deviation along with the increasing of the signal volume. According to these assessments, the

introduced Hurst exponent-based signal partition displays a stronger ability than the other two prevalent data processing approaches. Its effectiveness for the flank wear prediction could be illustrated as that the Hurst exponent partition simplifies the system frame with non-complex computation. The Hurst exponent helps accomplish the satisfying performance of the deep learning algorithm on the premise of not discarding the signal information.

#### 4.5 RUL prediction

After determining the superiority of the Hurst exponent partition method for the flank wear prediction, as the most ordinarily used tool life criteria, the flank wear further defined the cutting tool RUL based on the above prediction results. In general, the threshold of tool wear and tool life are determined by the actual application scenario and demand. Based on the dataset employed in this paper, the life of each cutting tool is defined as 315 cycles. In order to intuitively display the remaining life of the tool, the 315 corresponding actual flank wear values are expressed linearly by the remaining processable cycle of the tool. Furthermore, the remaining processable cycle of the predicted flank wear can be calculated as follows:

$$RUL_p^i = \frac{w_p^i \cdot RUL_a^i}{w_a^i} \quad (19)$$

where,  $RUL_a$  denotes the assigned actual remaining processable cycle, which is  $\{315, 314, \dots, 1\}$ .  $w_p$  denotes the predicted flank wear,  $w_a$  denotes the actual measured flank wear,  $i$  denotes the number of the cuts.

A polynomial regression fitting model is then constructed based on the sample data obtained by Equation 19 and the predicted flank wear for prediction of the RUL of the cutting tools. The regression function can be described as:

$$F(w) = \sum_{j=0}^k b_j \cdot w^j \quad (20)$$

where,  $F$  denotes the RUL of a cutting tool,  $w$  denotes the flank wear,  $b_j$  denotes the regression coefficients.

Figure 27 shows the comparison of the RUL prediction results of the partitioned and the un-partitioned datasets. And the RUL polynomial regression result of the partitioned datasets is depicted in Figure 28.

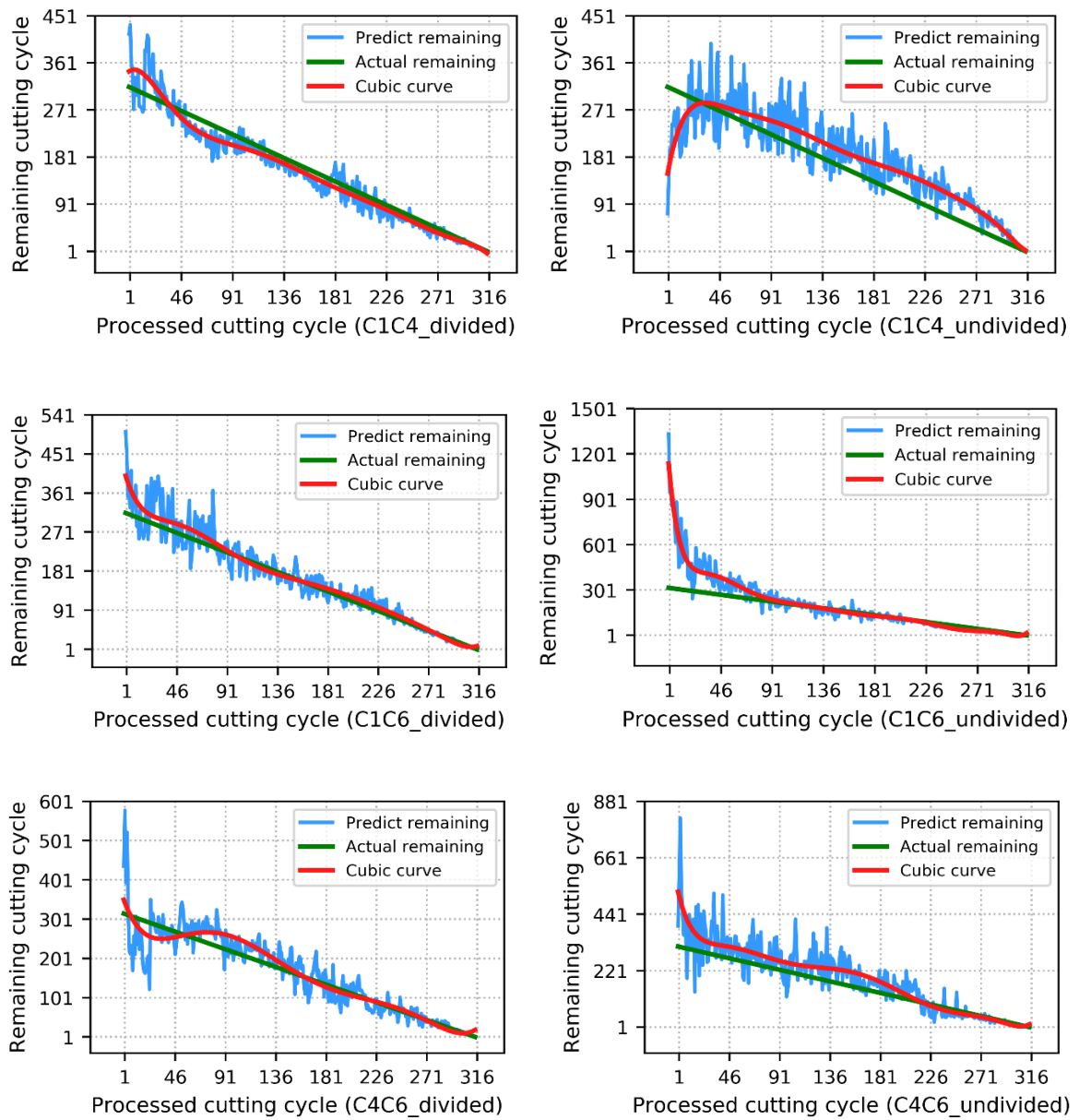


Figure 27. Predicted RUL and actual RUL of 3 cutting tools using partitioned and un-partitioned datasets.



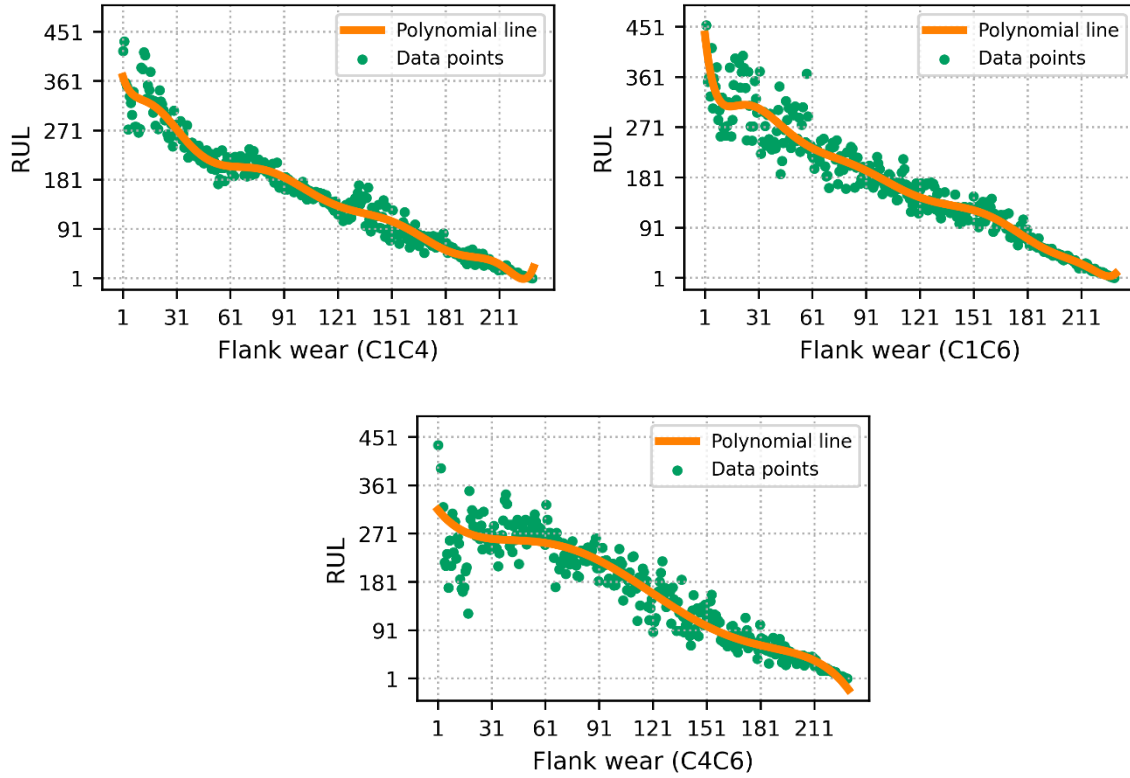


Figure 28. Polynomial regression of the RUL of 3 cutting tools using partitioned datasets.

It can be observed from Figure 27 that the predicted value of the RUL varies round the actual value, the overall trend persists unchanged.

Throughout the entire machining process, the predicted value always shows fluctuations near the actual value, the relative error is large at the beginning of the machining, and the predicted value of the tool life closely resembles the actual value during the subsequent machining.

With the tool wear deteriorates, the remaining machining cycle get easier to estimate. From the perspective of the segmented dataset C1C4, it can be seen that between the 1<sup>st</sup> cut and the 91<sup>th</sup> cut, the predicted cutting tool RUL fluctuates greatly and has certain errors. However, as the cutting continues until reaching the end of its life cycle (the 315<sup>th</sup> cutting cycles), the predicted value begins to approach the real value, and the prediction accuracy is guaranteed. In contrast, the prediction error of the un-partitioned dataset is still huge, in specific, the RUL of the cutting tool is underestimated in the interval from the 1<sup>st</sup> cut to the 20<sup>th</sup> cut, which may lead to early replacement of the cutting tool, and results in undesired waste.

After the 20<sup>th</sup> cut, the remaining life of the tool has been overestimated, which will cause the poor quality of the work-piece surface. Moreover, the prediction error of the undivided C1C6 dataset is more obvious that reaches the highest of 74%, and the undivided C4C6 dataset shows the overestimation in a long duration.

Additionally, the polynomial regression in Figure 28 displays that the model effectively fits the flank wear and the RUL, it can prove that the model has excellent performance in RUL prediction.

The accuracy of each regression prediction is assessed via the coefficient of determination ( $R^2$ ), which evaluates the extent of the model interpret and predict the result, the  $R^2$  value of 0.893, 0.898 and 0.856 are achieved by dataset C1C4, C1C6 and C4C6, respectively.

In the view of all results displayed above, the prediction accuracy of the dataset that partitioned by Hurst exponent presents significant improvement than that of the unsegmented dataset, and it illustrates the cutting tool RUL prediction system proposed in this paper has the potential and superior performance.

## 5. Conclusions

In order to enhance the prediction of the RUL of a cutting tool, a new methodology integrating the Hurst exponent and the hybrid CNN-LSTM algorithm is developed in a systematic means. In the research, a Hurst exponent-based method is developed to partition the signals of vibration, cutting force and AE collected along the lifecycles of a set of cutting tools. A hybrid CNN-LSTM algorithm is then designed to combine feature extraction, fusion and regression based on the multi-sourced and segmented signals to estimate the flank wear and RUL of cutting tools. A case study with a set of complex sensor signals was used to validate the developed methodology. To demonstrate the superior performance of the proposed methodology, a set of benchmarks with comparative algorithms, including CNN, LSTM, DNN, were conducted under the conditions of partitioned and un-partitioned signals. Additionally, the proposed CNN-LSTM algorithm was executed for prediction based on extracted features (in the time-, frequency- and time-frequency domains) and PCA dimensionally reduction. Results showed that, based on the case study in this research, the prediction accuracy of the proposed methodology reached 87.3%, which are significantly better than those of the benchmarking algorithms. Analyses on the results and observations were given in detail.

Future work will concentrate on the applications of the methodology on more case studies to prove the generality of the methodology. It will also investigate the possibility of further enhancing the accuracy of prediction via de-noising on signals and hybridisation with effective strategies.

## Declarations:

**Ethical Approval:** The authors claim that there are no ethical issues involved in this research.

**Consent to Participate:** All the authors consent to participate in this research and contribute to the research.

**Consent to Publish:** All the authors consent to publish the research. There are no potential copyright/plagiarism issues involved in this research.

**Funding:** This research was partially sponsored by the SABOT project funded by the IUK funder (UK). The research was also sponsored by the National Natural Science Foundation of China (Project No. 51975444).

**Conflicts of interest/Competing interests:** It declares that there are no conflicts of interest/competing interests for this research.

**Availability of data and material:** It declares that no data or materials are available for this research.

**Code availability:** It declares that codes are not available for this research.

**Authors' contributions:** Xiaoyang Zhang is responsible for idea and methodology development, algorithm implementation and validation, manuscript writing; Xin Lu, Weidong Li and Sheng Wang are responsible for supervision, idea and methodology discussion, algorithm check and manuscript refinement; Apart from the above contributions, Weidong Li is also responsible for funding support and manuscript finalisation.

## References:

- Q. An, Z. Tao, X. Xu, M. El Mansori and M. Chen, "A data-driven model for milling tool remaining useful life prediction with convolutional and stacked LSTM network", *Measurement*, vol. 154, p. 107461, 2020.
- R. Bhinge, J. Park, K. Law, D. Dornfeld, M. Helu and S. Rachuri, "Toward a Generalized Energy Prediction Model for Machine Tools", *Journal of Manufacturing Science and Engineering*, vol. 139, no. 4, 2016.
- T. Bogaerts, A. Masegosa, J. Angarita-Zapata, E. Onieva and P. Hellinckx, "A graph CNN-LSTM neural network for short and long-term traffic forecasting based on trajectory data", *Transportation Research Part C: Emerging Technologies*, vol. 112, pp. 62-77, 2020.
- P. Borys, "Long term Hurst memory that does not die at long observation times—deterministic map to describe ion channel activity", *Chaos, Solitons & Fractals*, vol. 132, p. 109560, 2020.
- G. Carneiro et al., *Deep Learning and Data Labeling for Medical Applications*. 2016.
- H. Chen, N. Hu, Z. Cheng, L. Zhang and Y. Zhang, "A deep convolutional neural network based fusion method of two-direction vibration signal data for health state identification of planetary gearboxes", *Measurement*, vol. 146, pp. 268-278, 2019.
- X. Chen, F. Kopsaftopoulos, Q. Wu, H. Ren and F. Chang, "A self-adaptive 1D convolutional neural network for flight-state identification", *Sensors*, vol. 19, no. 2, p. 275, 2019.
- H. Cheng, H. Chen, Z. Li and X. Cheng, "Ensemble 1-D CNN diagnosis model for VRF system refrigerant charge faults under heating condition", *Energy and Buildings*, vol. 224, p. 110256, 2020.
- J. Choi and B. Lee, "Combining LSTM network ensemble via adaptive weighting for improved time series forecasting", *Mathematical Problems in Engineering*, vol. 2018, pp. 1-8, 2018.

- H. Ismail Fawaz, G. Forestier, J. Weber, L. Idoumghar and P. Muller, "Deep learning for time series classification: a review", *Data Mining and Knowledge Discovery*, vol. 33, no. 4, pp. 917-963, 2019.
- S. Guan, H. Pang, W. Song and Z. Kang, "Cutting tool wear recognition based on MF-DFA feature and LS-SVM algorithm", *Transactions of the Chinese Society of Agricultural Engineering*, vol. 34, no. 14, pp. 61-68, 2018.
- Y. He et al., "Application of CNN-LSTM in Gradual Changing Fault Diagnosis of Rod Pumping System", *Mathematical Problems in Engineering*, vol. 2019, pp. 1-9, 2019.
- C. Herff and D. Krusienski, "Extracting features from time series", *Fundamentals of Clinical Data Science*, pp. 85-100, 2018.
- B. Kaya, C. Oysu, H. Ertunc and H. Ocak, "A support vector machine-based online tool condition monitoring for milling using sensor fusion and a genetic algorithm", *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, 226(11), pp. 1808-1818, 2012.
- T. Kim and S. Cho, "Predicting residential energy consumption using CNN-LSTM neural networks", *Energy*, vol. 182, pp. 72-81, 2019.
- M. Knight and M. Nunes, "Long memory estimation for complex-valued time series", *Statistics and Computing*, vol. 29, no. 3, pp. 517-536, 2018.
- W. Lee, M. Ratnam and Z. Ahmad, "Detection of chipping in ceramic cutting inserts from workpiece profile during turning using fast Fourier transform (FFT) and continuous wavelet transform (CWT)", *Precision Engineering*, vol. 47, pp. 406-423, 2017.
- K. Li, Y. Wu, Y. Nan, P. Li and Y. Li, "Hierarchical multi-class classification in multimodal spacecraft data using DNN and weighted support vector machine", *Neurocomputing*, vol. 259, pp. 55-65, 2017.
- T. Li, Z. Zhao, C. Sun, R. Yan and X. Chen, "Multi-scale CNN for Multi-sensor Feature Fusion in Helical Gear Fault Detection", *Procedia Manufacturing*, vol. 49, pp. 89-93, 2020.
- Z. Liu, J. Du, M. Wang and S. Ge, "ADCM: attention dropout convolutional module", *Neurocomputing*, vol. 394, pp. 95-104, 2020.
- H. Lotfalinezhad and A. Maleki, "TTA, a new approach to estimate Hurst exponent with less estimation error and computational time", *Physica A: Statistical Mechanics and its Applications*, vol. 553, p. 124093, 2020.
- L. Low, S. Yan, Y. Kwan, C. Tan and J. Thumboo, "Assessing the validity of a data driven segmentation approach: A 4 year longitudinal study of healthcare utilization and mortality", *PLOS ONE*, vol. 13, no. 4, p. e0195243, 2018.
- G. Martínez-Arellano, G. Terrazas and S. Ratchev, "Tool wear classification using time series imaging and deep learning", *International Journal of Advanced Manufacturing Technology*, vol. 104, no. 9-12, pp. 3647-3662, 2019.

- S. Mohanty, K. Gupta and K. Raju, "Hurst based vibro-acoustic feature extraction of bearing using EMD and VMD", *Measurement*, vol. 117, pp. 200-220, 2018.
- S. Opalka, B. Stasiak, D. Szajerman and A. Wojciechowski, "Multi-channel convolutional neural networks architecture feeding for effective EEG mental tasks classification", *Sensors*, vol. 18, no. 10, p. 3451, 2018.
- D. Rajamani, B. Esakki, P. Arunkumar and R. Velu, "Fuzzy logic-based expert system for prediction of wear rate in selective inhibition sintered HDPE parts", *Materials Today: Proceedings*, vol. 5, no. 2, pp. 6072-6081, 2018.
- R. Rusinek and M. Borowiec, "Stability analysis of titanium alloy milling by multiscale entropy and Hurst exponent", *European Physical Journal Plus*, vol. 130, no. 10, 2015.
- G. Serin, B. Sener, A.M. Ozbayoglu, H.O. Unver, "Review of tool condition monitoring in machining and opportunities for deep learning", *International Journal of Advanced Manufacturing Technology*, 109, pp. 953-947, 2020.
- S. Shankar, T. Mohanraj and A. Pramanik, "Tool condition monitoring while using vegetable based cutting fluids during milling of Inconel 625", *Journal of Advanced Manufacturing Systems*, 18(4), pp. 563-581, 2019.
- W. Sun et al., "An intelligent gear fault diagnosis methodology using a complex wavelet enhanced convolutional neural network", *Materials*, vol. 10, no. 7, p. 790, 2017.
- J. Wu, Y. Su, Y. Cheng, X. Shao, C. Deng and C. Liu, "Multi-sensor information fusion for remaining useful life prediction of machining tools by adaptive network based fuzzy inference system", *Applied Soft Computing*, vol. 68, pp. 13-23, 2018.
- X. Wu, Y. Liu, X. Zhou and A. Mou, "Automatic identification of tool wear based on convolutional neural network in face milling process", *Sensors*, vol. 19, no. 18, p. 3817, 2019.
- C. Xu, J. Dou, Y. Chai, H. Li, Z. Shi and J. Xu, "The relationships between cutting parameters, tool wear, cutting force and vibration", *Advances in Mechanical Engineering*, vol. 10, no. 1, p. 168781401775043, 2018.
- R. Yamashita, M. Nishio, R. Do and K. Togashi, "Convolutional neural networks: an overview and application in radiology", *Insights into Imaging*, vol. 9, no. 4, pp. 611-629, 2018.
- Y. Yang et al., "Research on the milling tool wear and life prediction by establishing an integrated predictive model", *Measurement*, vol. 145, pp. 178-189, 2019.
- S. Yousefi, M. Zohoor, "Effect of cutting parameters on the dimensional accuracy and surface finish in the hard turning of MDN250 steel with cubic boron nitride tool, for developing a knowledge base expert system", *International Journal of Mechanical and Materials Engineering*, 14(1), 2019.

- J. Yu, S. Liang, D. Tang and H. Liu, "A weighted hidden Markov model approach for continuous-state tool wear monitoring and tool life prediction", *International Journal of Advanced Manufacturing Technology*, vol. 91, no. 1-4, pp. 201-211, 2016.
- H. Zang, L. Liu, L. Sun, L. Cheng, Z. Wei and G. Sun, "Short-term global horizontal irradiance forecasting based on a hybrid CNN-LSTM model with spatiotemporal correlations", *Renewable Energy*, vol. 160, pp. 26-41, 2020.
- C. Zhang, X. Yao, J. Zhang and H. Jin, "Tool condition monitoring and remaining useful life prognostic based on a wireless sensor in dry milling operations", *Sensors*, vol. 16, no. 6, p. 795, 2016.
- Y. Zhang, N. Qin, D. Huang and K. Liang, "Fault diagnosis of high-speed train bogie based on deep neural network", *IFAC-PapersOnLine*, vol. 52, no. 24, pp. 135-139, 2019.
- R. Zhao, R. Yan, Z. Chen, K. Mao, P. Wang and R. Gao, "Deep learning and its applications to machine health monitoring", *Mechanical Systems and Signal Processing*, vol. 115, pp. 213-237, 2019.
- Y. Zhou, W. Xue, "Review of tool condition monitoring methods in milling processes", *International Journal of Advanced Manufacturing Technology*, 96, pp. 2509–2523, 2018.
- A. Zhu et al., "Exploring a rich spatial–temporal dependent relational model for skeleton-based action recognition by bidirectional LSTM-CNN", *Neurocomputing*, vol. 414, pp. 90-100, 2020.
- J. Zhu, H. Chen and W. Ye, "A hybrid CNN–LSTM network for the classification of human activities based on micro-doppler Radar", *IEEE Access*, vol. 8, pp. 24713-24720, 2020.