



This is a repository copy of *Intelligent real-time tool life prediction for a digital twin framework*.

White Rose Research Online URL for this paper:

<https://eprints.whiterose.ac.uk/226044/>

Version: Published Version

Article:

Dominguez-Caballero, J. orcid.org/0000-0001-7329-9929, Ayvar-Soberanis, S. orcid.org/0000-0002-1899-9743 and Curtis, D. orcid.org/0000-0001-6402-6996 (2025) Intelligent real-time tool life prediction for a digital twin framework. *Journal of Intelligent Manufacturing*. ISSN 0956-5515

<https://doi.org/10.1007/s10845-025-02606-4>

Reuse

This article is distributed under the terms of the Creative Commons Attribution (CC BY) licence. This licence allows you to distribute, remix, tweak, and build upon the work, even commercially, as long as you credit the authors for the original work. More information and the full terms of the licence here:

<https://creativecommons.org/licenses/>

Takedown

If you consider content in White Rose Research Online to be in breach of UK law, please notify us by emailing eprints@whiterose.ac.uk including the URL of the record and the reason for the withdrawal request.



eprints@whiterose.ac.uk
<https://eprints.whiterose.ac.uk/>



Intelligent real-time tool life prediction for a digital twin framework

Javier Dominguez-Caballero¹ · Sabino Ayvar-Soberanis¹ · David Curtis¹

Received: 3 October 2024 / Accepted: 2 April 2025
© The Author(s) 2025

Abstract

A key challenge in the machining manufacturing industry is real-time tool wear prediction, as conventional methods rely on conservative tool changes, causing premature replacement or excessive wear that risks failure, part damage, or poor surface quality. Monitoring and predicting the wear condition of a cutting tool is key to guarantee the cutting quality and saving costs. This study presents an AI-driven digital twin framework for real-time tool life prediction to address these limitations by integrating multiple modules. These modules include an on-machine direct inspection system, a seamless connectivity integration module for real-time data management, and a deep learning module for tool wear prediction. Long Short-Term Memory networks were trained, optimised and tested on a milling dataset to then deploy onto a real-time implementation of the digital twin framework. A comprehensive design of experiments (DOE) was used to validate the real-time tool life prediction framework of a dynamic milling toolpath strategy of a Ti-6Al-4 V alloy. The models were able to predict tool maximum flank wear based on sensor data from the machining tests DOE with RMSE of 33.17 μm , whilst the real-time implementation yielded a minimum of RMSE of 119.36 μm . These results motivate further research for enabling real-time closed-loop control for a future digital twin system implementation.

Keywords Tool wear · Deep learning · Digital twin · Real-time · Computer vision

Introduction

The industrial landscape has witnessed a transformative impact from the introduction and advances in artificial intelligence (AI) in recent years. The ever-growing volume of sensor data from CNC machines has enabled the development of data-driven models, accelerating the transition towards “smart manufacturing” (Moore et al., 2020). This trend is mirrored in machining research, where a significant focus lies on creating AI models utilising machine learning (ML) and deep learning (DL) for diverse applications. These applications include tool wear prediction, tool breakage detection, energy consumption optimisation, and surface roughness prediction (Yang et al., 2023). Existing research underlines the potential of data-driven models to achieve high accuracy, flexibility, and robustness when equipped with appropriate data capture methods (Liu et al., 2023a).

Within the CNC machining process, the cutting tool plays a pivotal role. Tool wear and tool life are critical factors for evaluating a material’s machinability. They provide measurable information about a material’s resistance to machining. Different materials can be compared based on their tool wear or tool life under identical machining conditions. Tool life is defined as the time or amount of material removed before tool wear reaches a specific limit. However, tool life can be influenced by many factors beyond just the material itself, such as cutting speed, cutting strategy, tool type, tool engagement and wider system conditions (e.g. machine tool, fixture, and lubrication). This makes it challenging to directly compare machinability across different studies (Liao et al., 2024). Furthermore, tool health directly influences the quality and efficiency of the machined products (Ambhore et al., 2015). Tool wear progressively accumulates over time, ultimately culminating in tool failure. This failure can significantly hinder production efficiency by compromising the accuracy of the machined product or even causing scrapped parts. Conversely, precise tool wear monitoring can enable increased cutting speeds and reduced production downtime (Kurada

✉ Javier Dominguez-Caballero
j.dominguez-caballero@sheffield.ac.uk

¹ The University of Sheffield Advanced Manufacturing Research Centre, Sheffield, UK

& Bradley, 1997). In critical applications, tool life and processes are often predetermined. To mitigate the effects of tool breakage, tools in industrial processes are intended to be replaced before failure occurs. However, a lack of confidence in the current condition of the tool can lead to conservative decision-making. Conventional replacement strategies can rely on fixed intervals based on operators' subjective experience. This approach can result in premature tool replacement, increasing costs and downtime, or delayed replacement, which negatively affects workpiece quality and raises production costs (Zhou & Xue, 2018). Therefore, for optimal production efficiency and product quality, accurate tool wear prediction is paramount. Traditional tool wear monitoring methods are categorised as either direct or indirect (Ambhore et al., 2015). Direct methods, which involve physical measurement of the tool through techniques like microscopy, can be intrusive and impractical for continuous, real-time monitoring in a production environment. Indirect methods, rely on analysing sensor signals such as cutting forces, vibrations, and temperature during the machining process (Dimla & Lister, 2000). Even though, these methods are impacted by the complex, non-linear relationships between sensor data and the actual tool wear state (Wang et al., 2022), they can provide a comprehensive insight into the physical system.

A digital twin is a virtual representation of a physical system that continuously learns, and updates based on sensor data streamed from its physical counterpart. In machining, a digital twin acts as a virtual counterpart, mirroring the physical machining process in real-time. This digital representation is driven by a continuous stream of data flowing directly from the CNC machining tool. It encompasses real-time sensor data capturing various aspects of the machining operation, such as cutting forces, vibration levels, and spindle motor power (Ward et al., 2021). Additionally, a digital twin can integrate data-driven digital models such as DL. These models are constantly fed in real-time using the information received from the CNC machine and any other relevant sensory equipment. This continuous flow of data allows the digital twin to not only reflect the current state of the machining process, but also predict future behaviour and potential outcomes. This real-time synchronisation between the physical and digital worlds can be a powerful tool for optimising machining processes, predicting issues like tool wear, and making informed decisions for improved production efficiency and product quality.

While Artificial Neural Networks (ANNs) have proven effective in various applications due to their ability to learn and model non-linear relationships (Sun et al., 2022), however, their limitations in handling long-term dependencies in time series data pose a challenge for tool condition monitoring (TCM) in machining. Long Short-Term Memory (LSTM) networks, a specific type of Recurrent Neural Network (RNN), address this limitation. LSTMs are designed

to specifically learn these long-term dependencies within sequential data, making them particularly well-suited for tasks like tool wear prediction in machining. By overcoming the vanishing gradient problem that hinders traditional RNNs, LSTMs can effectively capture the complex relationships between sensor data collected over time and the corresponding tool wear state (Hochreiter & Schmidhuber, 1997). This capability allows LSTMs to predict tool wear with high accuracy, leading to improved production efficiency and reduced costs associated with tool failures. Marani et al. (2021), Sayyad et al. (2022), and Kumar et al. (2022) investigated the use of an LSTM and BiLSTM models to predict tool flank wear, whilst Chen et al. (2019), Duan et al. (2023) and Li et al. (2022) combined convolutional neural networks (CNN) and bidirectional LSTMs (BiLSTM) to classify or predict tool wear in machining. Whilst their approaches showed a promising framework to enable real-time predictions, these studies relied on controlled environments with limited parameter variation, reducing their applicability to dynamic industrial settings. Zhang et al. (2022) proposed a method to predict tool wear and cutting forces in micro milling. They combined a data-driven approach through LSTMs to predict tool wear and influence a physics-based model to improve cutting force prediction accuracy. The method considers factors like tool run-out and tool wear for a more realistic model. Guo et al. (2022) implemented a pyramid LSTM auto-encoder for tool wear monitoring in high-speed machining (HSM). This model leverages the periodic nature of cutting signals to achieve accurate prediction, even under complex and changing conditions. The design allowed the model to learn from unlabelled data and reduced computational cost compared to traditional methods. Despite achieving high accuracy in offline scenarios, these studies lack real-time implementation and often depend on extensive post-processing, limiting their practical deployment.

Regarding the research of real-time tool life prediction, Liu et al. (2023b, 2024) propose a framework for TCM during machining. This framework looks to incorporate physical knowledge of the machining process to improve the accuracy of TCM using a model frequency analysis coupled with a nonlinear autoregressive with exogenous input (NARX) network. Their experimental results showed that this model can be applied to a range of machining scenarios in real-time. However, the system had computational speed limitations in real-time implementation, requiring further research for industrial applications. While the existing research demonstrates the promise of DL models for tool wear prediction, several limitations hinder their widespread industrial adoption. A significant portion of the validation process relies on controlled machining tests with straight cutting operations. These tests often involve repeated trials with similar parameters or a limited combination of settings. This approach may

not adequately reflect the real-world complexities encountered in diverse machining scenarios. Additionally, the implementation of DL models is primarily focused on offline settings, showcasing their potential for online predictions such as the work form Chen et al. (2019), but still limited to a tool wear classification approach. However, there is a scarcity of examples demonstrating full integration and real-time functionality within industrial machining processes; and very few implementing data-driven deep learning techniques capable of generalising the complex machining behaviour.

The current paper proposes a novel digital twin framework using DL models for real-time tool wear prediction in milling operations. By using DL models within this framework, the system can continuously analyse sensor signals in real-time and predict tool wear indirectly with high accuracy. The DL's ability to model complex relationships between input data and desired outputs makes it particularly well-suited for this task (Liu et al., 2023a). Furthermore, the proposed digital twin framework would allow the model to continuously learn and adapt to changing machining conditions, such as variations in workpiece material, cutting parameters, or tool geometry by also integrating a direct in-situ vision inspection system to initially collect tool wear data and later serve as a source of verification, and in future work, continuous learning. A continuous learning capability would ensure the model's predictions remain accurate over extended periods, even under fluctuating production conditions. This approach offers significant advantages over existing solutions that are commercialised and are available to industry such as ARTIS Marposs (2022), Nordmann Tool Monitoring (Nordmann, 2017), Montronix (2022), and Caron Engineering TMAC (Caron, 2022); which primarily rely on comparing sensor readings to fixed thresholds or analysing trends to identify critical events like tool breakage or collisions. Hence, these commercial systems offer limited insights into the actual state of the tool condition. By enabling real-time predictions, the current framework approach could allow both, the use of cutting tools to their full extent, as well as proactive interventions. During operations where cutting tools are replaced continuously assuming the end of life has been reached only by experience or predefined machining time (Zhou & Xue, 2018), this system could help extend the use of the tools. Conversely, when the predicted tool wear reaches a pre-defined threshold, the system can trigger an alert for tool replacement, minimising unplanned downtime and production losses. Overall, this real-time tool wear prediction digital twin framework using deep learning has the potential to significantly improve production efficiency, reduce costs associated with scrapped parts and tool failures, and ultimately enhance product quality in CNC machining operations. To comprehensively evaluate the efficacy of the proposed digital twin framework, a DOE was employed. This DOE included a broad design space, specifically targeting

the expansion of the deep learning (DL) models' operational limits. The machining tests incorporated complex milling tools featuring variable pitch and helix angles. Additionally, the tests utilised Ti-6Al-4 V, a material commonly used in aerospace and medical applications. Furthermore, a complex dynamic milling toolpath, as encountered in blisk machining (a critical aerospace operation), was implemented. The data acquired during these trials served a dual purpose: training and validating the DL models. Subsequently, these models were deployed and tested in a fully integrated real-time implementation on a new machining trial that was not trained or tested, demonstrating good accuracy.

The novelty of this research lies in its real-time execution of AI-based tool wear prediction, addressing gaps in prior literature that rely on post-processed data rather than live inference. A large DOE with parameter variation enhances model robustness, while validation on unseen real-time data confirms its generalisation ability. These contributions to the advancement of digital twin technology for machining look to demonstrate a scalable and practical solution for real-world manufacturing environments.

Proposed framework and methods

Architecture

The framework of the intelligent real-time tool life prediction digital twin is illustrated in Fig. 1, which includes the development and integration of three main modules: direct in-situ inspection, real-time connectivity, and deep learning.

Direct inspection methodology

The direct inspection module is an in-situ vision-based system designed for on-machine tool wear measurements to capture wear progression. This system leverages image processing algorithms to quantify tool wear directly during the machining process, as depicted by Fig. 2. The workflow begins with the creation of a digital mask from an image of a new, unworn, tool. This mask serves as a reference for the original tool edge. Subsequently, the mask is aligned and subtracted from an image of the worn tool, effectively removing background features. Following background removal, a thresholding technique is employed to isolate the region of interest (ROI) corresponding to the worn area. Finally, image segmentation algorithms are utilised to delineate the wear profile, enabling the measurement of the maximum flank wear at various tool locations (i.e., bottom, middle, and top). The individual wear measurements from these locations are then averaged to generate a single value representing the average maximum flank wear across the entire tool length.

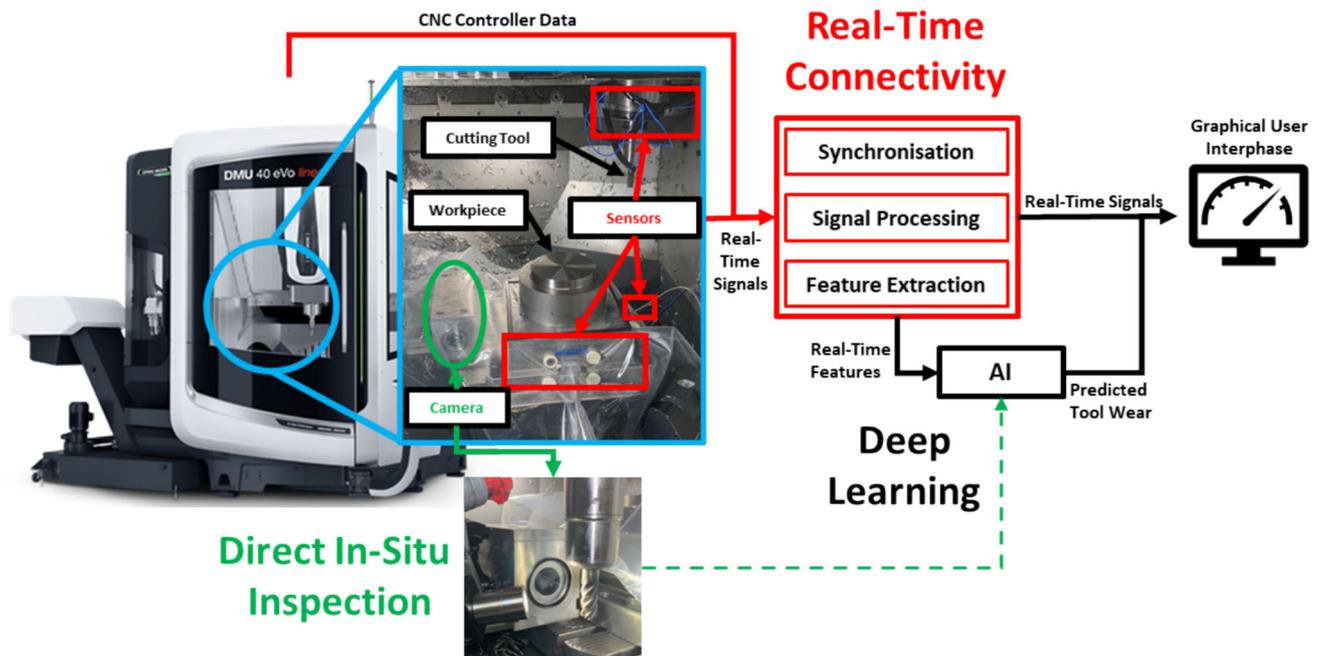


Fig. 1 Proposed intelligent digital twin framework for real-time tool life prediction

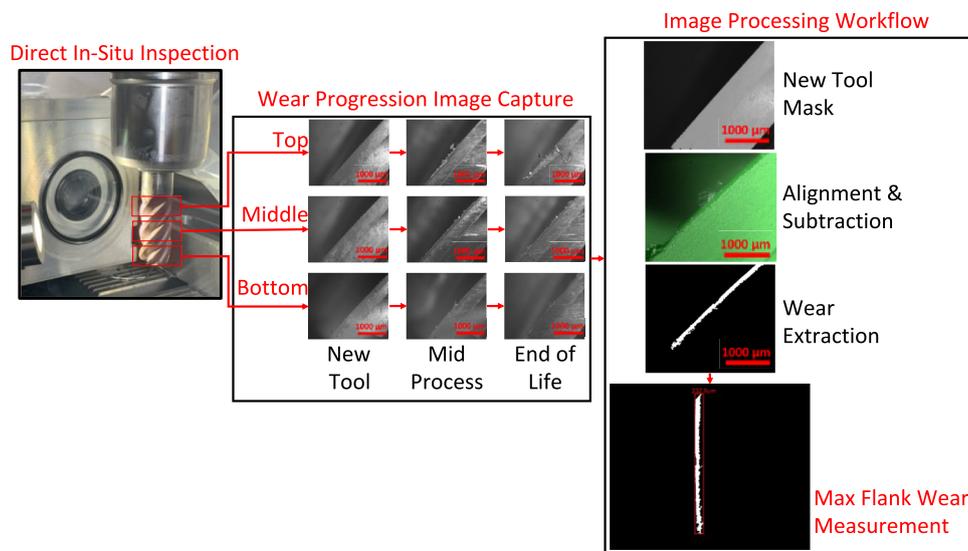


Fig. 2 Direct in-situ inspection and processing workflow

This framework functionality enables the collection of continuous tool wear measurements during a machining process to acquire the data to be used for training the DL models. Additionally, this system can continue to be relevant during the deployment of the system with the trained models to occasionally verify the predictions, help improve accuracy and enable continuous learning with development further work. This approach would ensure that the number of verification measurements is reduced to a minimum, impacting less the machining downtime to carry out this task.

Real-time connectivity methodology

The real-time connectivity module of the digital twin framework enables real-time data collection of the various sensors retrofitted to the CNC machine, along with data collected from the machine numerical control kernel (NCK). This module performs synchronisation of the various data streams and implements the data processing and feature extraction tasks. A software was created to collect the various data streams implementing parallel callback function cycles to

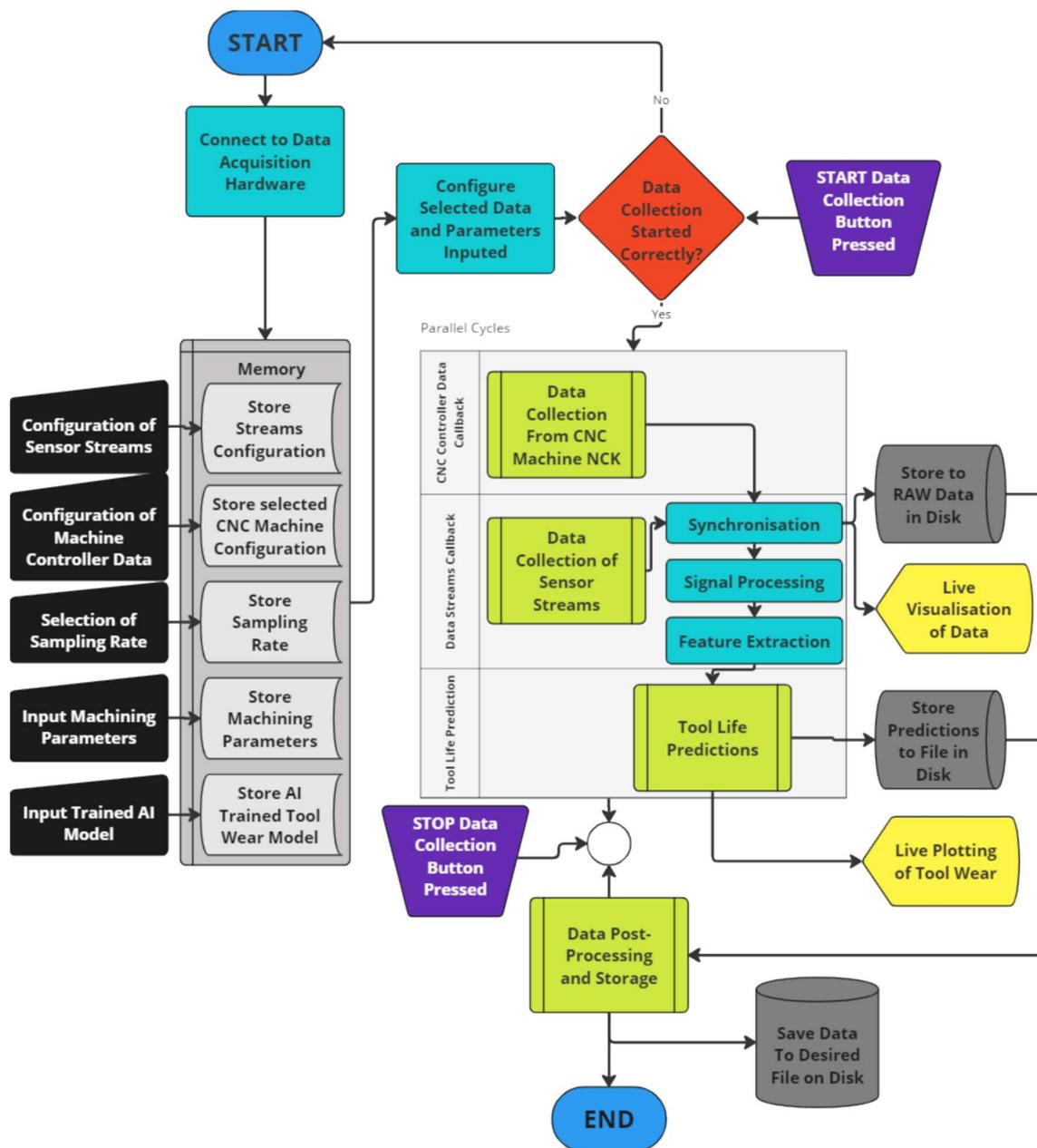


Fig. 3 Intelligent digital twin software and real-time connectivity flow diagram

collect the data and save internally. A separate callback function receives the processed real-time features to carry out predictions. The software functionality and the data flow are illustrated by the flow diagram shown in Fig. 3.

Data processing and feature extraction methodology

For the DL training tasks, the data captured during the machining testing required processing to isolate the signal from the non-cutting data to avoid any misleading signals from initial stages where the tool is not fully engaged and

potential material dimensional variabilities, as represented by Fig. 4. Similarly, filtering was done to avoid introducing cutting dynamic issues to the models as localised vibrations were observed in the mid-cuts, possibly due to low rigidity of the machine tool at those locations.

The segmented signals for each milling cut were recorded at different sampling rates (51,200 Hz and 333 Hz) and were subsequently subdivided into half a second sections (*i*). A commonly used feature extraction technique for AI-based tool wear prediction is to extract statistical features (Colantonio et al., 2021; Kumar et al., 2022; Moore et al.,

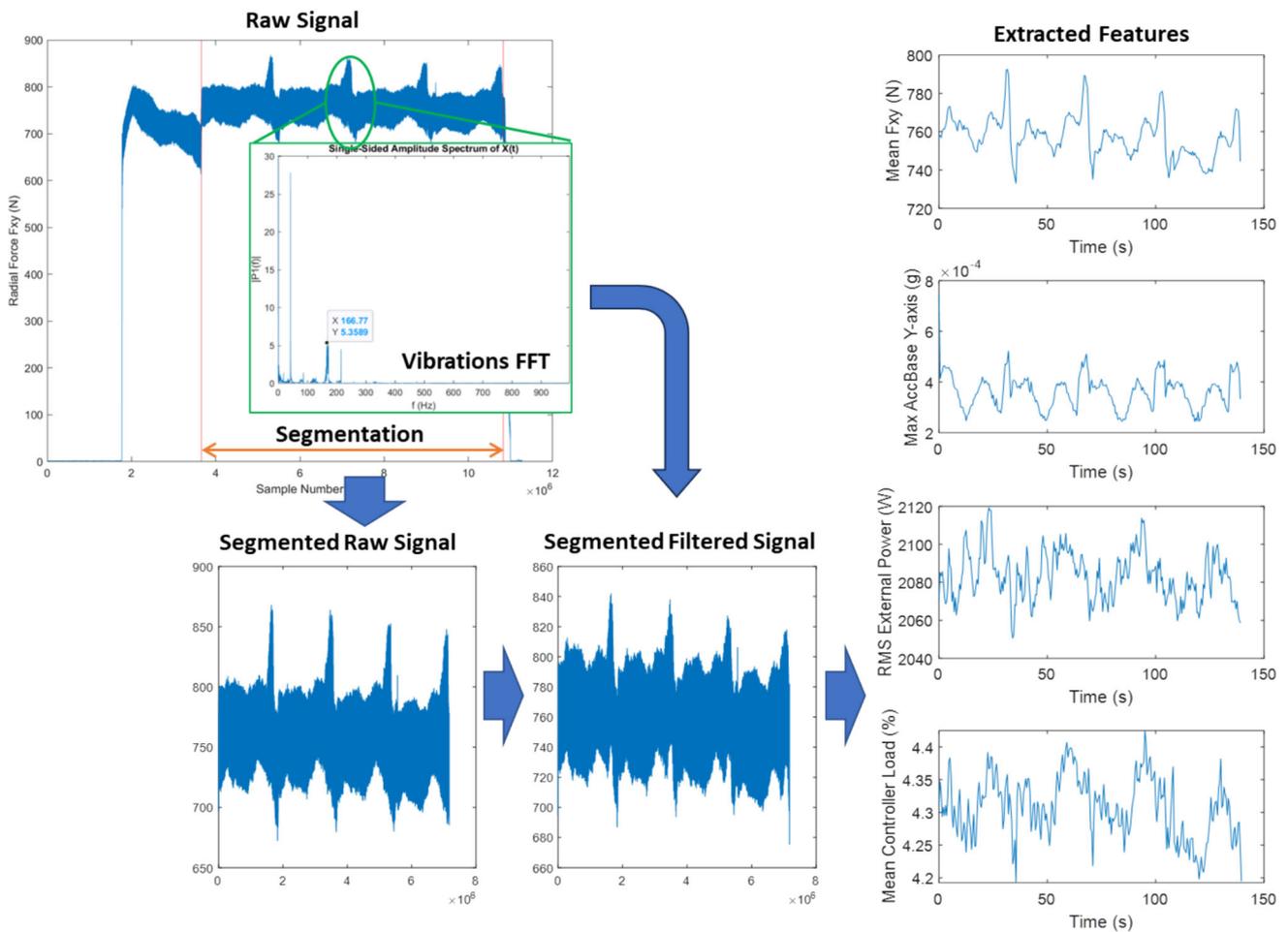


Fig. 4 Signal segmentation and processing

2020; Sayyad et al., 2022). Therefore, time-domain and time-frequency-domain statistical features were extracted from each section, as depicted by the extracted features in Fig. 4. The labelling was conducted based on extracted statistical features to ensure uniform representation of machining conditions. Table 1 shows the time-domain statistical features used, where x_i represents the current signal value and N represents the total number of samples in the evaluated section of the signal.

For the time-frequency-domain features, mean, standard deviation, skewness and kurtosis values were calculated from the signal spectral kurtosis ($K(f)$) using Eq. 9, where $S(t, f)$ is a short-time Fourier transform described by Eq. 10, $w(t - \tau)$ is the evaluated window, $x(t)$ represent the signal, τ represents the phase unwrapping on the time axes, and f is the frequency axes.

$$K(f) = \frac{\langle |S(t, f)|^4 \rangle}{\langle |S(t, f)|^2 \rangle^2} - 2 \tag{9}$$

Table 1 Statistical features equations

Statistical feature	Equation	Equation number
Maximum	$X_{max} = \max(x)$	(1)
Mean	$X_{mean} = \frac{1}{N} \sum_{i=1}^N x_i$	(2)
Standard deviation	$X_{std} = \sqrt{\frac{\sum_{i=1}^N (x_i - X_{mean})^2}{N}}$	(3)
Skewness	$X_{skew} = \frac{\sum_{i=1}^N (x_i - X_{mean})^3}{(N-1)X_{std}^3}$	(4)
Kurtosis	$X_{kurt} = \frac{\sum_{i=1}^N (x_i - X_{mean})^4}{(N-1)X_{std}^4}$	(5)
Peak to peak (P2P)	$X_{p2p} = \max(x_i) - \min(x_i)$	(6)
Root mean square (RMS)	$X_{rms} = \sqrt{\frac{1}{N} \sum_{i=1}^N x_i^2}$	(7)
Crest factor	$X_{crest} = \frac{X_{max}}{X_{rms}}$	(8)

$$S(t, f) = \int_{-\infty}^{+\infty} x(t)w(t - \tau)e^{1\pi ft} dt \tag{10}$$

Deep learning methodology

The DL module receives the real-time data features and carries out live predictions of tool wear, which could enable the machine operator to carry out decisions on the machining operations. This task is executed in parallel to the main data collection tasks to avoid issues with this or any other potential usages of the signals (e.g. data analytics, other simulation modules, etc.). When this parallel function finishes computing, the DL model outputs are pulled out and displayed onto the Graphical User Interface (GUI).

Long short-term memory (LSTM) networks

The LSTM networks are a type of recurrent neural networks suitable for time-series data with the architecture described in Fig. 5. These networks use LSTM blocks for each time step (t), where each block has initial hidden (h_{t-1}) and cell (c_{t-1}) states that get updated at each timestep and become inputs (h_t and c_t) for the next block, along with the timeseries (x), to generate a final output network state prediction (y_T). The main advantage of the LSTM networks is that the cell state provides the long-term memory of the network, whereas the hidden state has the ‘short-term’ memory from the previous output. Each LSTM block has the system of gates shown in Fig. 6 to carry out the network update by an *input*, *forget*, *update*, and *output gate* (Hochreiter & Schmidhuber, 1997).

Input Gate This gate deals with the control of the information flow that updates the cell state (long-term memory). This

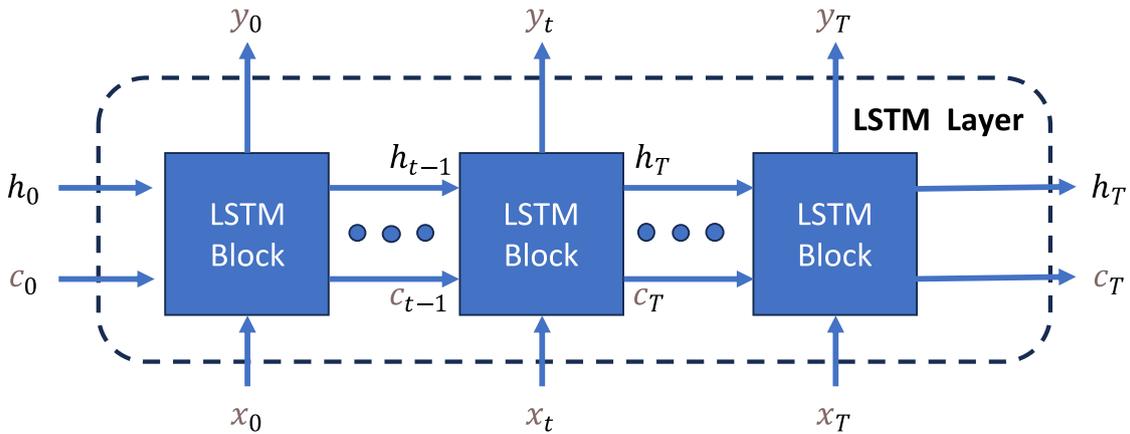


Fig. 5 LSTM layer architecture

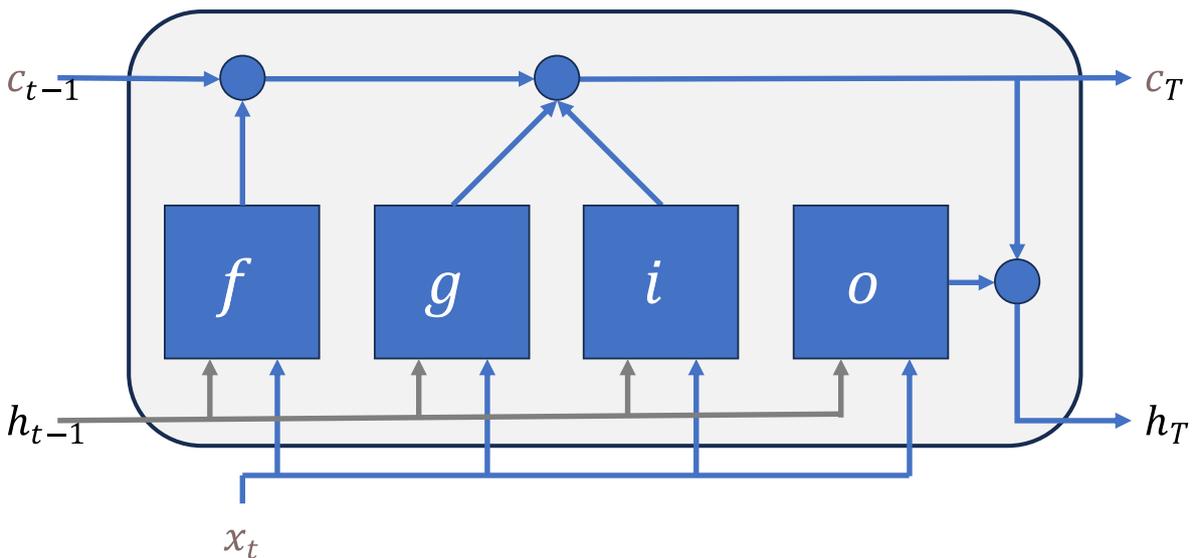


Fig. 6 LSTM block architecture

gate contains an ANN (i) with a sigmoid activation function (σ_S), allowing it to discriminate the relevance of any new data for re-training the network, as described by Eq. 11. Where W denotes the weights, R the recurrent weights, and b the biases for this gate; the subscript refers to the specific gate, which in this case is the input gate.

$$i_t = \sigma_S(W_i x_t + R_i h_{t-1} + b_i) \quad (11)$$

Forget Gate The *forget gate* (F) contains an ANN (i) with a sigmoid activation function trained with the new sample (x_t) and the previous hidden state (h_{t-1}) as inputs, and the output is a vector with values between 0 and 1 to discriminate the input relevance using Eq. 12. This output vector is multiplied elementwise by the previous cell state c_{t-1} and passed to the update gate, allowing the network to ‘forget’ or make cell components less influential for the subsequent steps, as described by Eq. 13.

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

$$F = f_t \odot c_{t-1} \quad (13)$$

Update Gate The *update gate* (U) determines what information gets stored in the cell state (long-term memory) of the network. The cell candidate (g) is updated by an ANN with a tanh activation function (σ_h) trained to combine previous hidden states with new data, providing an updated magnitude to the cell state but without any ‘relevance’ discrimination, as described by Eq. 14. This cell candidate is multiplied elementwise by the input gate output vector using Eq. 15, to then be added to the vector coming from the forget gate to create a new cell state (c_t), and thus update the ‘long-term memory’ of the network, as described by Eq. 16.

$$g_t = \sigma_h(W_g x_t + R_g h_{t-1} + b_g) \quad (14)$$

$$U = i_t \odot g_t \quad (15)$$

$$c_t = F + U \quad (16)$$

Output Gate The *output gate* (O) passes the new cell state directly to the next block and updates the hidden state (h). An output vector (o) is generated by an ANN using a sigmoid activation function. The hidden state, which also outputs the latest time step prediction (y_t), gets updated by an elementwise multiplication between the output vector and the new cell state that was passed through a tanh activation function to normalise the data between -1 and 1 using Eq. 17.

$$y_t = h_t = o_t \odot \sigma_h(c_t) \quad (17)$$

During the training of the LSTM network, one of the main parameters to be defined are the number of hidden units, which relates to the amount of information the network remembers between time steps or the hidden state. This is an important parameter to avoid overfitting. Furthermore, the LSTM network can be linked to ‘fully connected layers’ with a set of weights to further improve predictions for high-dimensional problems. The number of hidden units for these layers also require selecting to avoid overfitting.

Bayesian optimisation

Bayesian optimisation is widely used for ML and DL hyperparameter selection, as these networks can become difficult and expensive to optimise with other optimisation algorithms that rely on accessing the objective function’s derivatives (Frazier, 2018). Bayesian optimisation uses surrogate optimisation that consists of generating a surrogate function to approximate the objective function, which in this case is the DL network. Based on the surrogate function, promising minima points can be identified, and these regions are furtherly explored, and the surrogate function can be updated accordingly. This is done iteratively to learn more about the areas of interest and update the function until a suitable or global minima is found. The surrogate functions are commonly represented by Gaussian Process, which is a probability distribution over possible fitting functions (Snoek et al., 2015; Wu et al., 2019) described by Eq. 18 given a mean function (μ) and a covariance function ($k \rightarrow \mathbb{R}$).

$$f(x) \sim GP(\mu(x), k(x, x')) \quad (18)$$

The surrogate function can be updated with new information through the Bayesian process, and the Gaussian Process distribution can be used to find the global minima by being a cheaper objective function to optimise. This optimisation was used to iteratively train multiple networks whilst varying and optimising the desired model parameters, hyperparameters and architectures to achieve the results presented in “[Results and analysis](#)” section.

Data augmentation

Data augmentation is a widely used technique in DL to artificially expand the size and diversity of a training dataset by applying controlled transformations to existing data points. This process helps to improve the generalisation ability of DL models, making them more robust to inherited variability of unseen real-time data (Chawla et al., 2002; Fernández-Delgado et al., 2014; Shorten & Khoshgoftaar, 2019).

This technique is commonly applied to image data, which can include rotations, flips, cropping, colour jittering, and noise addition. These transformations simulate real-world

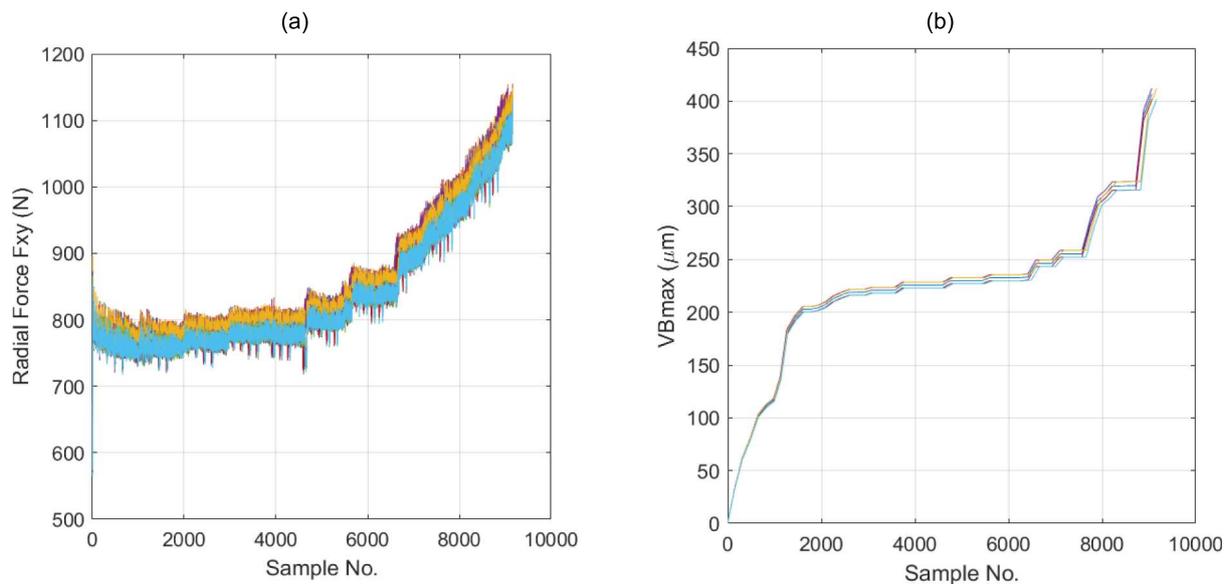


Fig. 7 Results on Test I for: **a** Variable noise on signals by 2.5%; **b** offset of 2.5% on tool wear

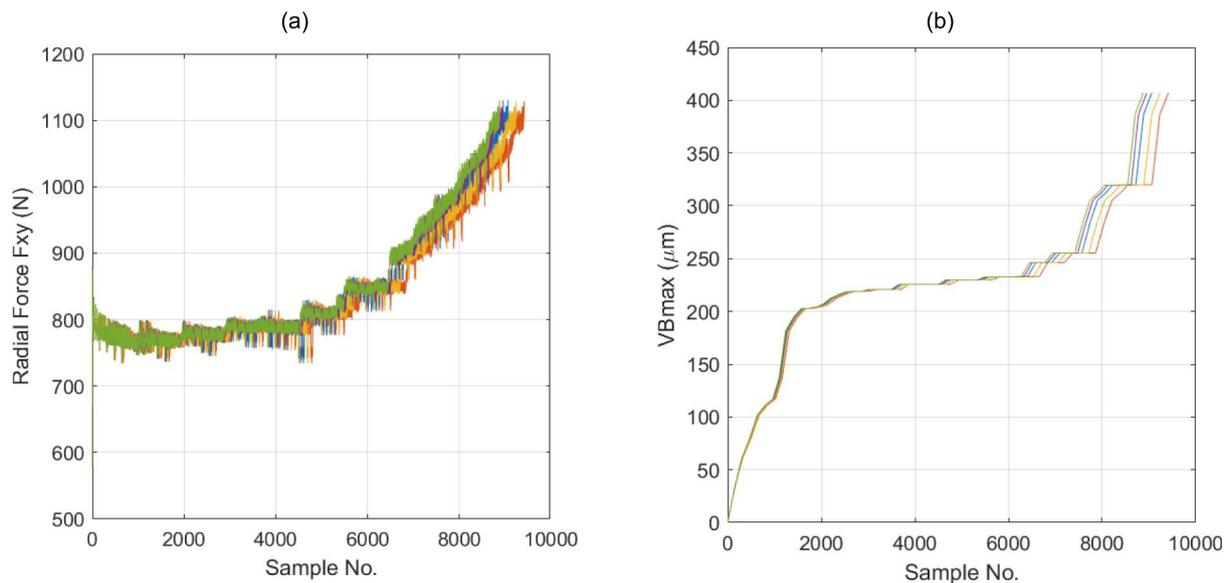


Fig. 8 Results of time warping of 5% on Test I for: **a** sensor signals; **b** tool wear profiles

variations that a model might encounter. Time series or sequence data can also be augmented to enhance robustness by simulating realistic signal variations, thereby improving generalisation while preventing over-sampling and redundancy. However, excessive augmentation can lead to unrealistic variability; therefore, empirically determined noise thresholds were implemented to balance robustness and accuracy. Other studies have demonstrated the efficacy of data augmentation methods, in enhancing AI models' robustness and accuracy in machining and industrial applications (Jiang et al., 2023; Martins et al., 2023).

Two techniques implemented to augment the data acquired during the machining testing are the following: **Noise addition** Artificially add noise to individual datapoints or the full signal to represent variability from sources such as sensor errors, environmental factors, etc. (Gao et al., 2023). **Time warping** Stretch or compress the signal in the time axis to represent temporal variations (Petitjean et al., 2011).

For the noise addition technique, the original sensor signals and the tool wear profiles were first offset in full by a noise factor of $\pm 2.5\%$ to keep the data inside a 95% confidence level, which equates to 2σ of a normally distributed process noise. Following this, the original sensor signals were

affected by a noise factor of $\pm 2.5\%$ on each data point (across time), as shown in Fig. 7a. However, it should be noted that this was not applied to the tool wear profiles, which were only affected with the full signal offset factor of $\pm 2.5\%$, as shown in Fig. 7b, given that this would have been less representative of real variability in tool wear data, which is a discrete direct measurement rather than a signal prone to variability. Finally, for the time warping technique, new signals were produced by extending and compressing the original data by a random percentage of $\pm 5\%$ of the original length, also to keep inside the 95% confidence level, as shown in Fig. 8. This higher percentage was used to try to account for the stochastic variability on tool life, which may happen in practice between different experimental trials using identical tools and identical machining conditions. Each technique generated six new signals by applying three positive and three negative factor values to ensure data diversity and balance. The data labels remained unchanged, as these augmentation methods only introduced minor transformations preserving wear progression trends.

Experimental method

Experimental setup

The experimental testing was carried out in a 5 axis DMG DMU 40 eVo linear CNC milling machine tool as presented in Fig. 9a. The workpiece was mounted onto an adapter plate, which was in turn mounted onto a plate dynamometer as depicted by Fig. 9b. Figure 9c illustrates the data acquisition setup, including the equipment used and its connectivity. The sensing devices were used to monitor the process, which included an external spindle power monitoring kit PPC-3, three tri-axial accelerometers PCB 356A02 (two on the spindle, one on the dynamometer), and a Kistler dynamometer 9255C connected to a Kistler charge amplifier 5070. These sensors were then connected to a National Instruments (NI) cDAQ-9178 chassis using modules NI-9239 for the dynamometer, NI-9234 for the accelerometers, and NI-9201 for the power kit. The NI chassis that was plugged into the data acquisition PC via USB. Machine controller data was also collected via Transmission Control Protocol/Internet Protocol (TCP/IP) using an ethernet connection; this stream of data was enabled through a Heidehain LSV2 connector. The configuration of the inspection system consisted of an area scan camera (Basler acA2040-25gm), and lighting equipment (SVL SXW30-W), as well as a power over the ethernet (PoE) ethernet connection from the camera to a laptop. The sensor data was collected at a sampling rate of 51,200 Hz, and the machine controller data was collected at the NCK rate of 333 Hz.

For the machining tests, 14 cylindrical billets of a dimension of $\text{Ø}200 \text{ mm} \times 100 \text{ mm}$ of Ti-6Al-4 V material, hot rolled to AMS 4928 were machined. The tooling selected for the machining trials was an OSG UVX-Ti-5FL endmill, which had variable pitch and helix angles detailed in Table 2. Blaser Vasco 7000 flood coolant was used during the machining trials, with a concentration of $\sim 8\%$.

Machining testing

A dynamic milling toolpath strategy can help maintain a constant cutting engagement and tool load throughout the machining process. This approach typically involves adjusting the tool's trajectory and feed rate dynamically, based on the geometry of the workpiece, to maximise material removal while minimising tool wear and reducing cutting forces. This approach was used during machining testing to vary the cutting speed (V_c) and radial depth of cut (a_e) whilst keeping a constant maximum chip thickness (h_{ex}), which for these tests was kept at $60 \mu\text{m}$. The cutting strategy consisted of 24 cuts of equal volume (98.98 cm^3) per billet, where six cuts were carried out on each billet of the four levels of the workpiece material, as depicted in Fig. 10. A maximum flank wear (VB_{max}) is commonly used to define the limits where a tool is regarded as worn. According to ISO 3685, 1993 standards (ISO, 1993), a limit of 0.3 mm can be used for uniform wear; therefore, a limit of 0.3 mm was used as the threshold of tool failure during the machining trials.

A full-factorial design of experiments as outlined by Table 3, was implemented where the centre point of the tests was repeated twice. This design was also complemented with extra tests to increase the amount of data for the deep learning modelling.

Results and analysis

Using the in-situ inspection methodology, the tool wear was measured after each cut. Figure 11 shows an example of a flute of the tool used in Test A for the start, middle and end of life wear in the top, middle and bottom section of the tool. To ensure repeatability in tool wear measurements, the CNC machine's spindle encoder was used to precisely and consistently position the tool for imaging. Additionally, each tool was cleaned using compressed air before image capture to remove coolant residues and swarf, preventing measurement inaccuracies.

The measured tool wear is presented in Fig. 12, where each line corresponds to a test of the DOE. A review was done after the tests to carry out an inspection of the data and the expected trends. Inspection of the tool wear profiles from the tests revealed a clear correlation between tool life and the selected machining parameters, including cutting speed, radial depth

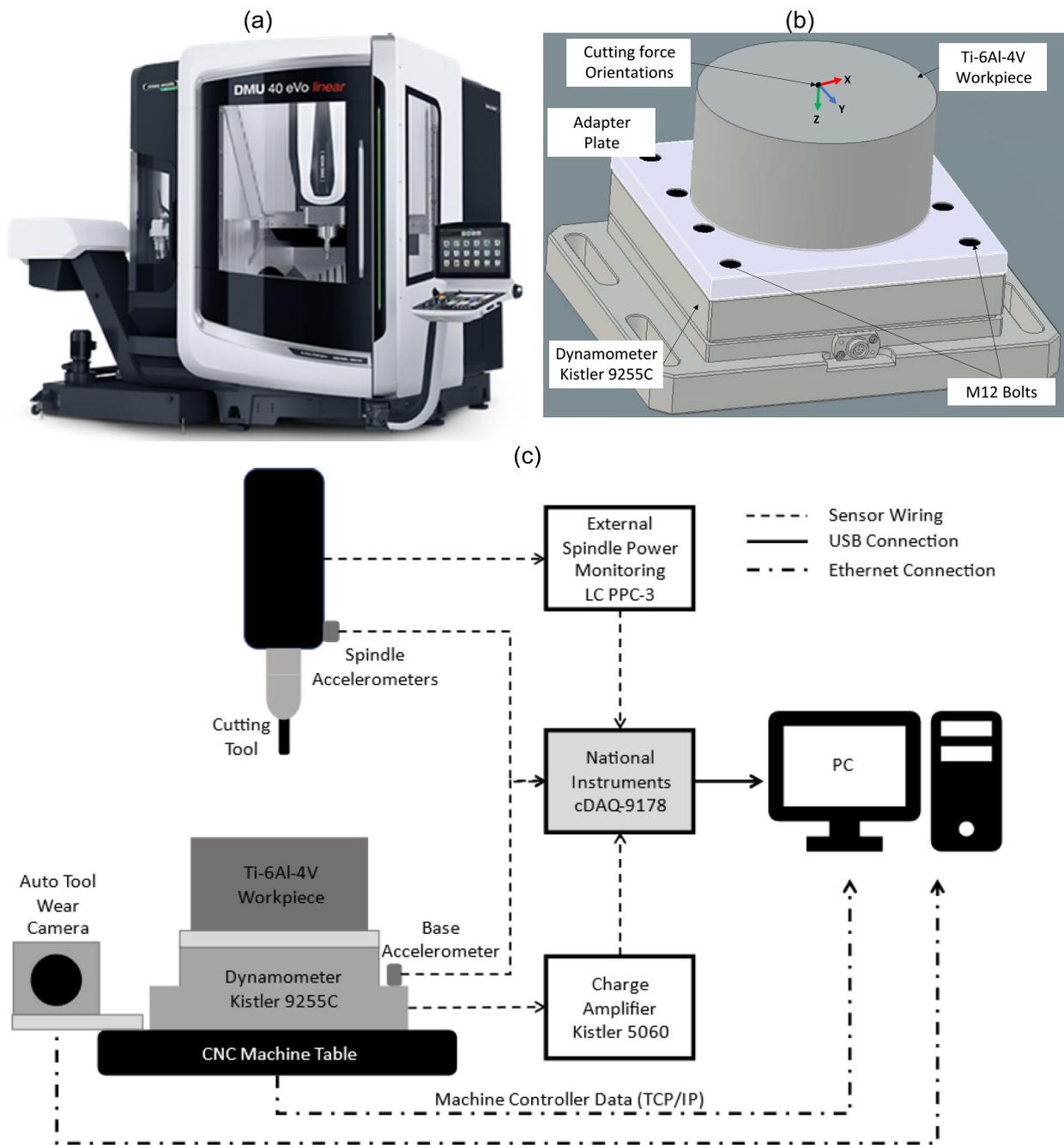


Fig. 9 Experimental setup: a DMG DMU eVo 40 linear; b workpiece fixturing setup; c data acquisition setup

Table 2 Tooling specifications

Tool code	Diameter (mm)	Corner radius (mm)	Helix angles (°)	Pitch angles (°)
UVX-TI-5FL 16XR3X48	16	3	41/42/43	83/68/64/83/62

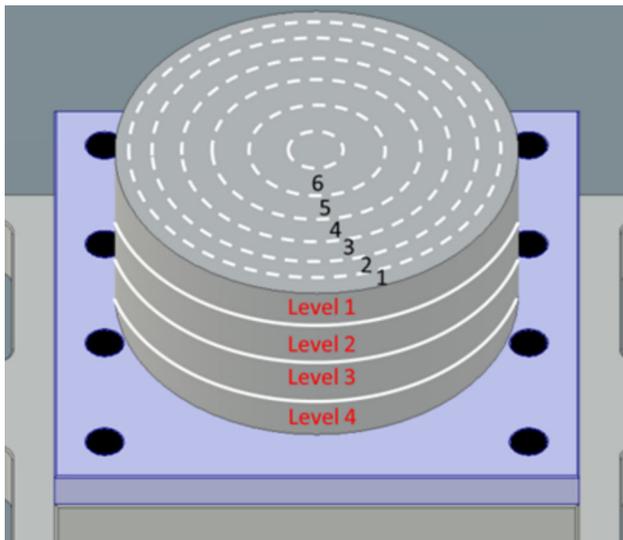


Fig. 10 Machining strategy diagram showing workpiece cuts and levels

of cut, and feed rate. Additionally, when comparing the tool wear profiles with the mean of the XY resultant forces (F_{xy}) calculated Eq. 19, there was a good correlation as observed in Fig. 13.

$$F_{xy} = \sqrt{F_x^2 + F_y^2} \quad (19)$$

The repeated tests at the centre point of the DOE, test E and test I, had very close tool life profiles (Fig. 14a) and mean F_{xy} (Fig. 14b), indicating very low variability during testing. This is also evidenced by the Pearson's correlations in Fig. 15a for the tool wear profiles, and Fig. 15b for the mean xy , where both show very high positive correlation coefficients, which is a measure of a linear relationship or

dependence between pairs of corresponding datapoints in the timeseries, with p-values below 0.05.

The DOE test data was split into nine training (tests A, C, D, E, F, G, H, I and J) and one validation (test B) datasets. The data augmentation techniques were then applied, and these were able to increase the nine original testing datasets illustrated in Fig. 16, into the 2205 datasets as shown in Fig. 17. This was also the case for the validation dataset, which was augmented from 1 to 245 datasets.

The Bayesian optimisation was applied to the training datasets systematically adjusting hyperparameters such as LSTM hidden units, fully connected layer nodes, batch size, and dropout rates, minimising the mean absolute error (MAE) for the validation dataset using Eq. 20, where \hat{y}_i correspond to the i th iteration trained network predictions and y_i corresponds to the actual tool wear values. Figure 18 shows the optimisation results for 150 optimisation iterations due computational efficiency and diminishing improvements in validation loss, as additional improvements beyond this value were found minimal.

$$MAE = \frac{1}{N} \sum_{i=1}^N |\hat{y}_i - y_i| \quad (20)$$

Following the optimisation process, the four best performing networks were selected. Their training and validation performance was evaluated using the root mean squared error (RMSE) metric (Eq. 21). Table 4 summarises the networks architecture and their results.

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N |\hat{y}_i - y_i|^2} \quad (21)$$

Figure 19 presents examples of the prediction results for the augmented validation dataset using the four networks.

Table 3 Experimental test for tool wear machining trials

	Cutting speed V_c (m/min)	Feed per tooth f_z (mm/tooth)	Spindle speed n (rpm)	Feed rate V_f (mm/min)	Radial DoC a_e (mm)	Axial DoC a_p (mm)
Test A	130	0.084	2586	1086	1.6	20
Test B	140	0.075	2785	1044	1.6	20
Test C	150	0.069	2984	1034	4	20
Test D	130	0.075	2586	970	1.6	20
Test E	140	0.075	2785	1044	3.2	20
Test F	150	0.084	2984	1254	2.4	20
Test G	140	0.069	2785	965	4	20
Test H	130	0.069	2586	896	4	20
Test I	140	0.075	2785	1044	3.2	20
Test J	150	0.100	2984	1492	1.6	20

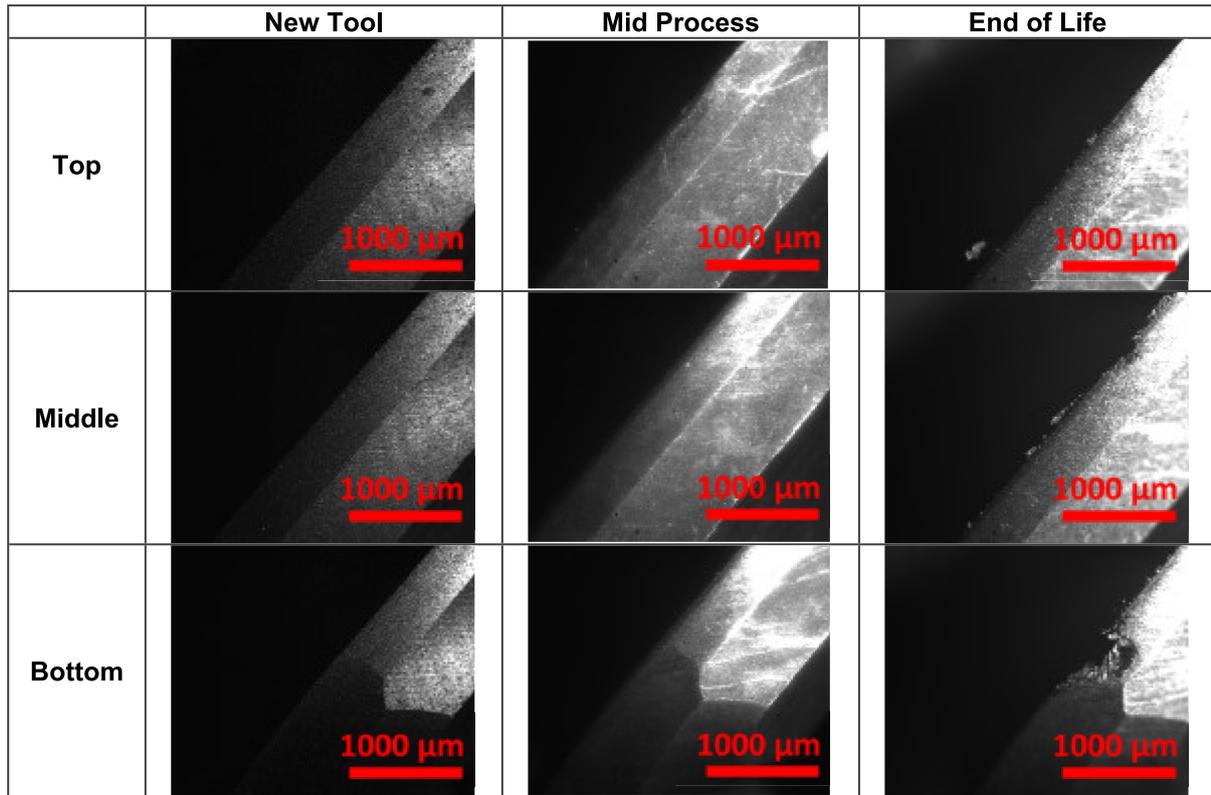


Fig. 11 Tool wear images of top, middle and bottom of flute 1 in Test A for new, mid process and end of life conditions

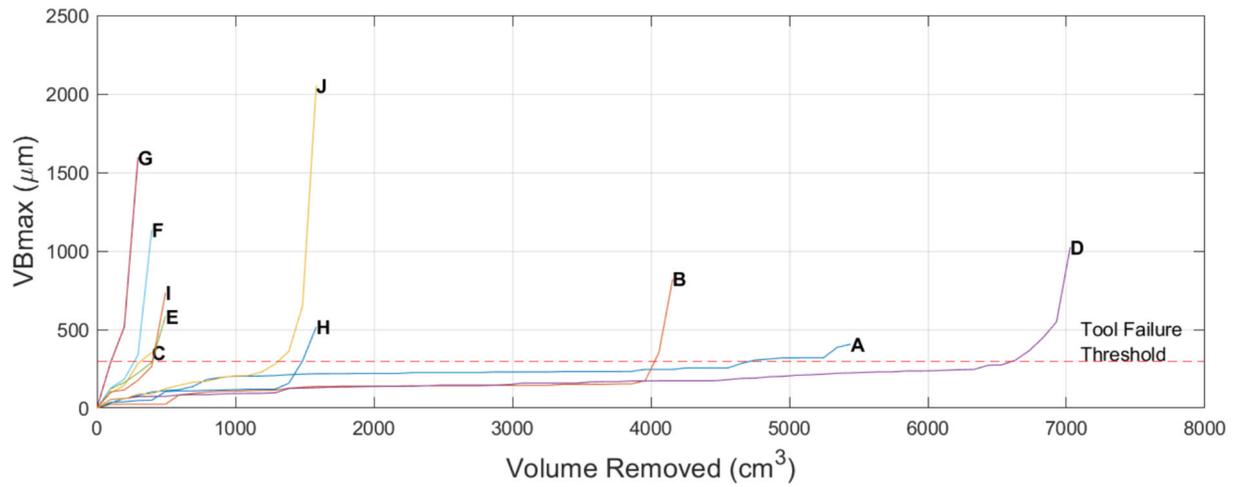


Fig. 12 Tool wear profiles captured during the DOE machining tests

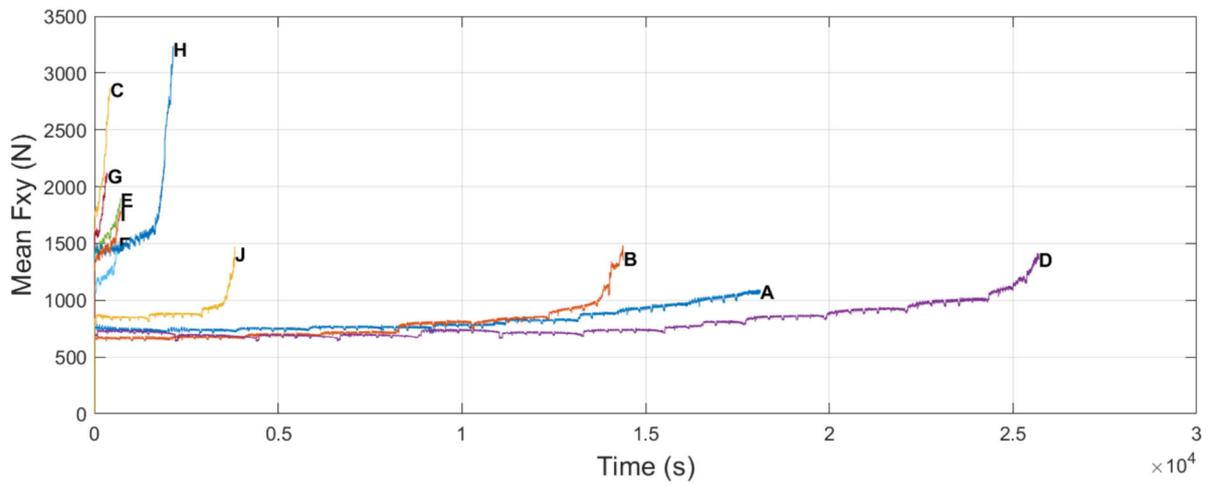


Fig. 13 Mean XY resultant forces (Fxy) of the machining tests

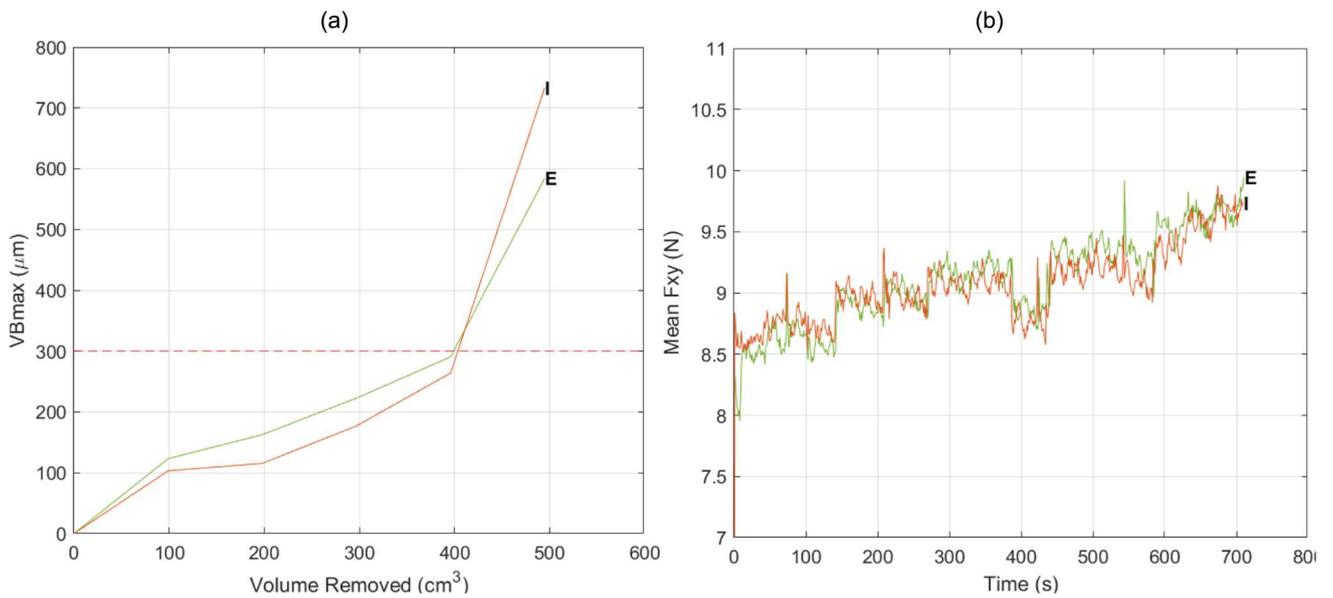


Fig. 14 Results from tests E and I: a tool wear profile; b mean XY resultant forces (Fxy)

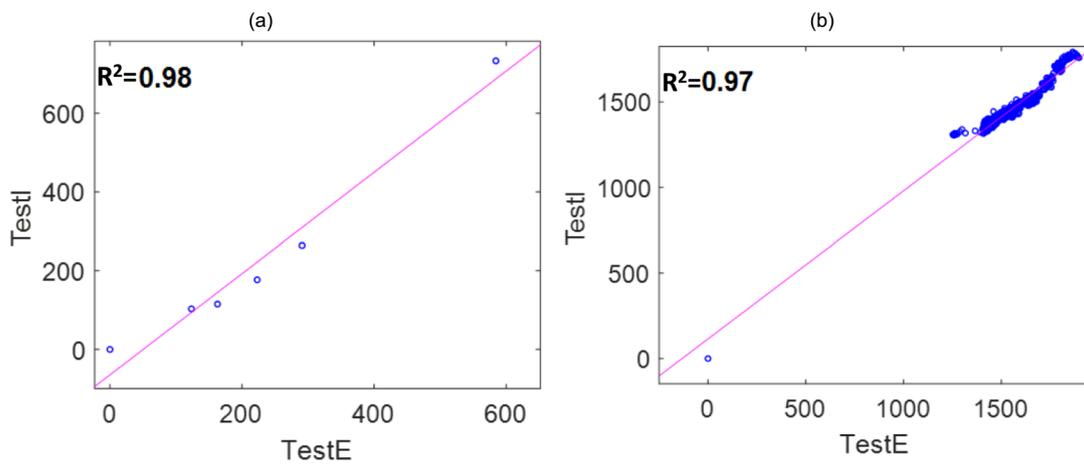


Fig. 15 Correlation coefficients between tests E and I for: a tool wear; b mean XY resultant forces (Fxy)

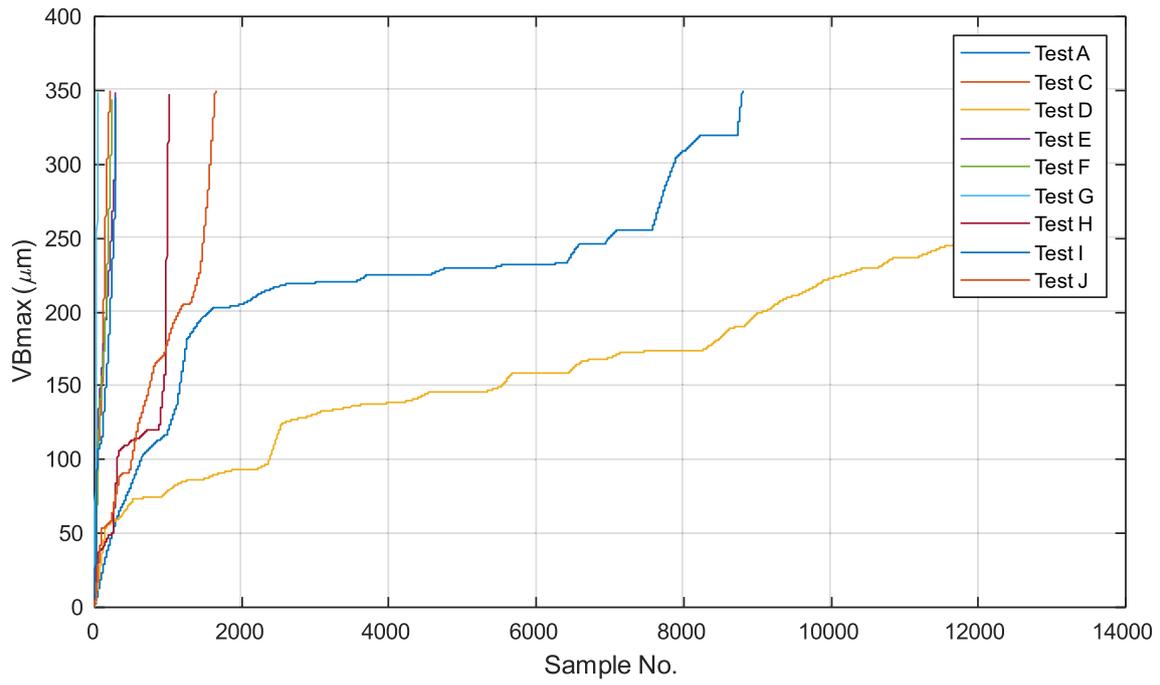


Fig. 16 Original training datasets

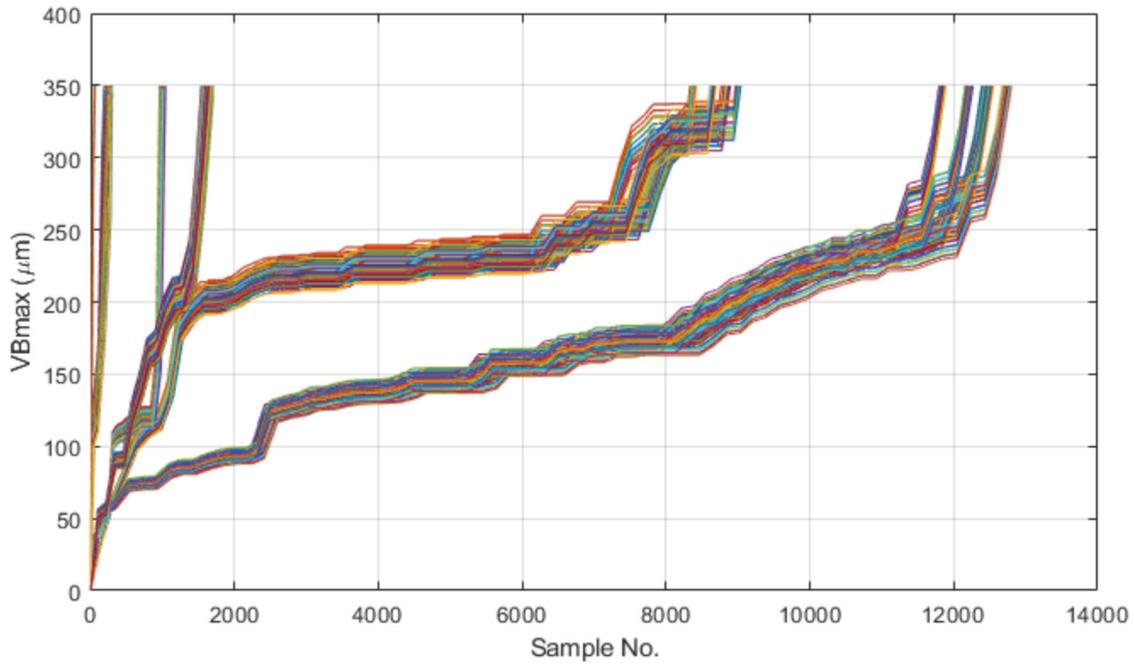


Fig. 17 Augmented training datasets

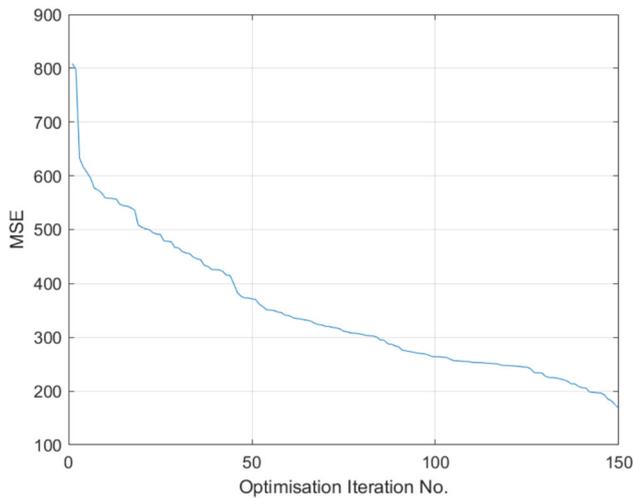


Fig. 18 LSTM validation MAE Bayesian optimisation results

These results showed accurate tool wear predictions with RMSE values as low as $33.17 \mu\text{m}$ for the dataset acquired during the DOE machining testing. The networks were later deployed for a real-time implementation of the digital twin framework on a new machining test. This test replicated the cutting parameters of Test B from the DOE, enabling a comparison between the offline and real-time results. The framework executed data collection, processing, and feature extraction in real-time, feeding live features to the networks in a parallel manner (as depicted previously in Fig. 3). Subsequently, the networks predicted wear, updated their internal state based on the latest data point, and displayed the current wear prediction to the user interface as shown in Fig. 20. Figure 21 includes the predictions from the four networks, while Table 5 presents the corresponding prediction RMSE values.

The results for the real time prediction revealed reasonably accurate with RMSE values as low as $119.36 \mu\text{m}$, hence the predictions followed closely the tool wear trend and captured the inflection point of the rapid wear stage leading to failure. Interestingly, most networks predicted slightly higher wear values compared to the measured data. This discrepancy could be attributed to a potentially more aggressive initial wear stage in the real-time test compared to the initial DOE machining tests dataset. This could lead to a higher steady wear stage inflection point, creating a steeper wear trajectory.

These challenges are anticipated when dealing with real-time sensor data, which may be susceptible to noise and the inherent stochastic nature of the machining process; as evidenced by the correlation results previously described for the centre points of the DOE machining tests, which were highly correlated repetitions, but still not identical. Nonetheless, these results successfully validate the proof-of-concept for the digital twin framework, demonstrating promising outcomes.

The scope of this research was to evaluate the feasibility of deploying AI models for real-time tool wear prediction in an industrially relevant setting. While the study provides insights into wear progression trends, it is not a fundamental analysis of wear mechanisms. Future work should focus on improving the accuracy of the deep learning models and ensure these provide better real-time predictions and integrating additional process monitoring techniques to enhance predictive accuracy. Similarly, future work should look at the full integration of the in-situ inspection system to enable the continuous learning during the real-time implementation of the system, whenever tools are inspected in-situ and mid process to verify the predictions.

Table 4 Training and validation prediction RMSE (mean and max) results for DOE machining testing

Network No	Architecture				Training RMSE (μm)	Validation RMSE (μm)
	LSTM hidden units	Fully connected hidden units	Drop out rate	Batch size		
1	51	108	0.2466	6	7.75	33.17
2	130	95	0.2494	5	17.87	65.5
3	132	150	0.227	5	28.16	49.47
4	136	93	0.2003	6	28.21	52.96

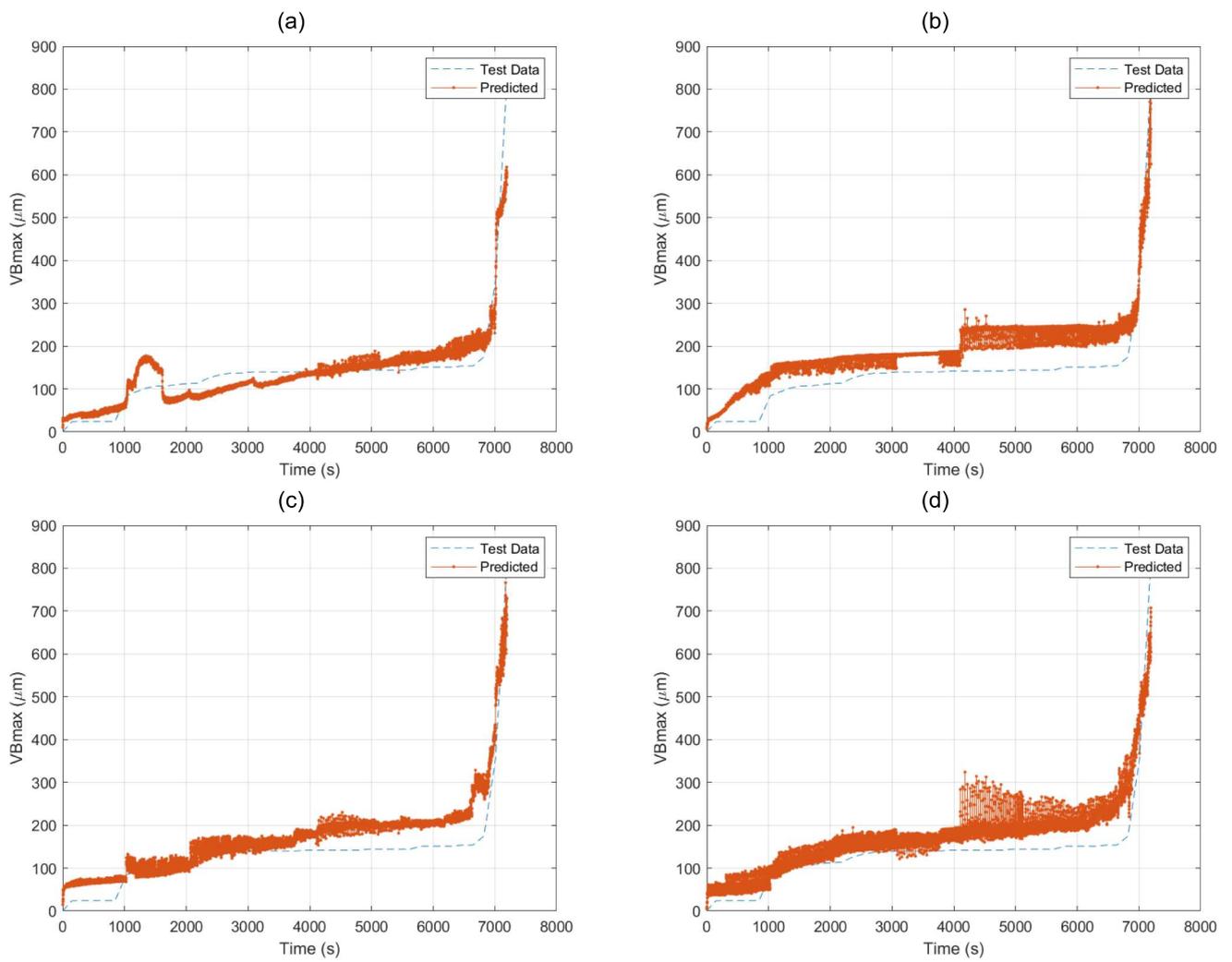


Fig. 19 Validation prediction results for the top four networks on the DOE machining testing: **a** network 1; **b** network 2; **c** network 3; **d** network 4

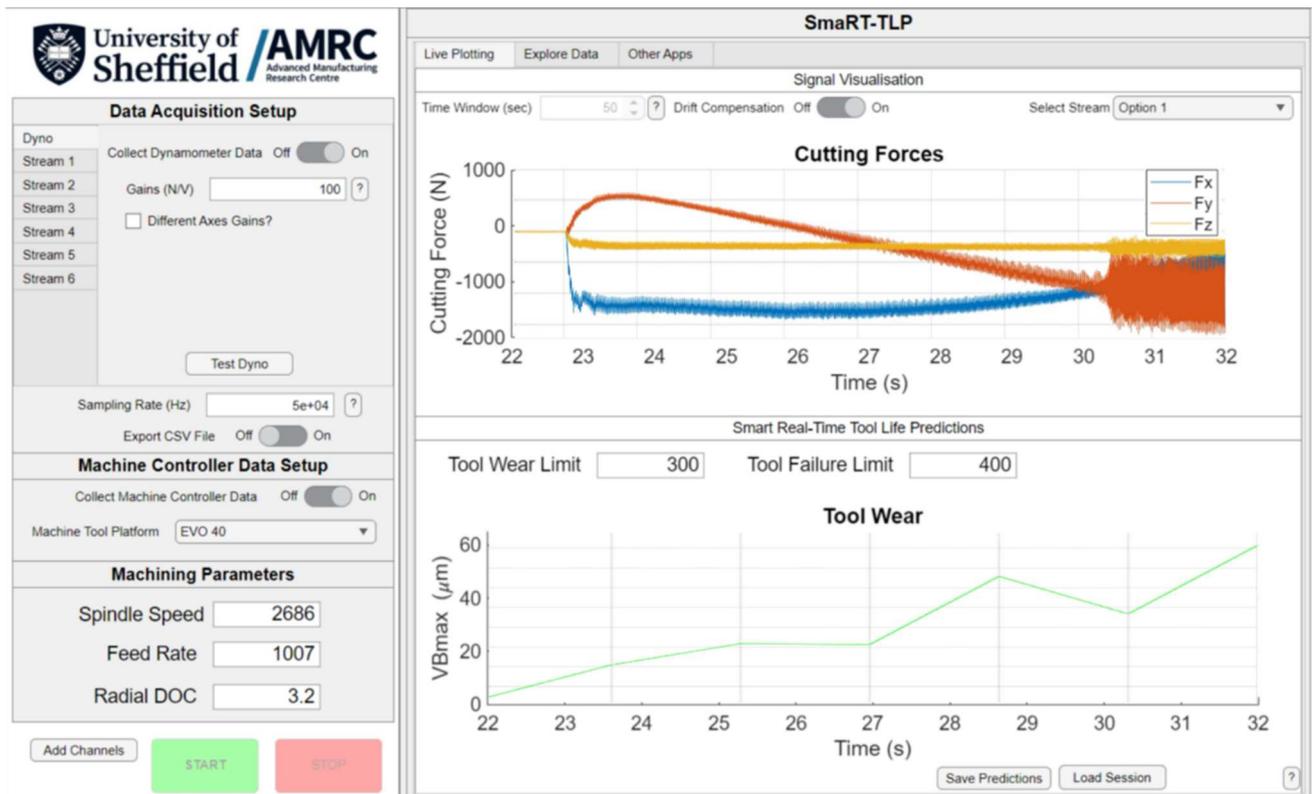


Fig. 20 Digital twin framework user interface during real-time machining testing

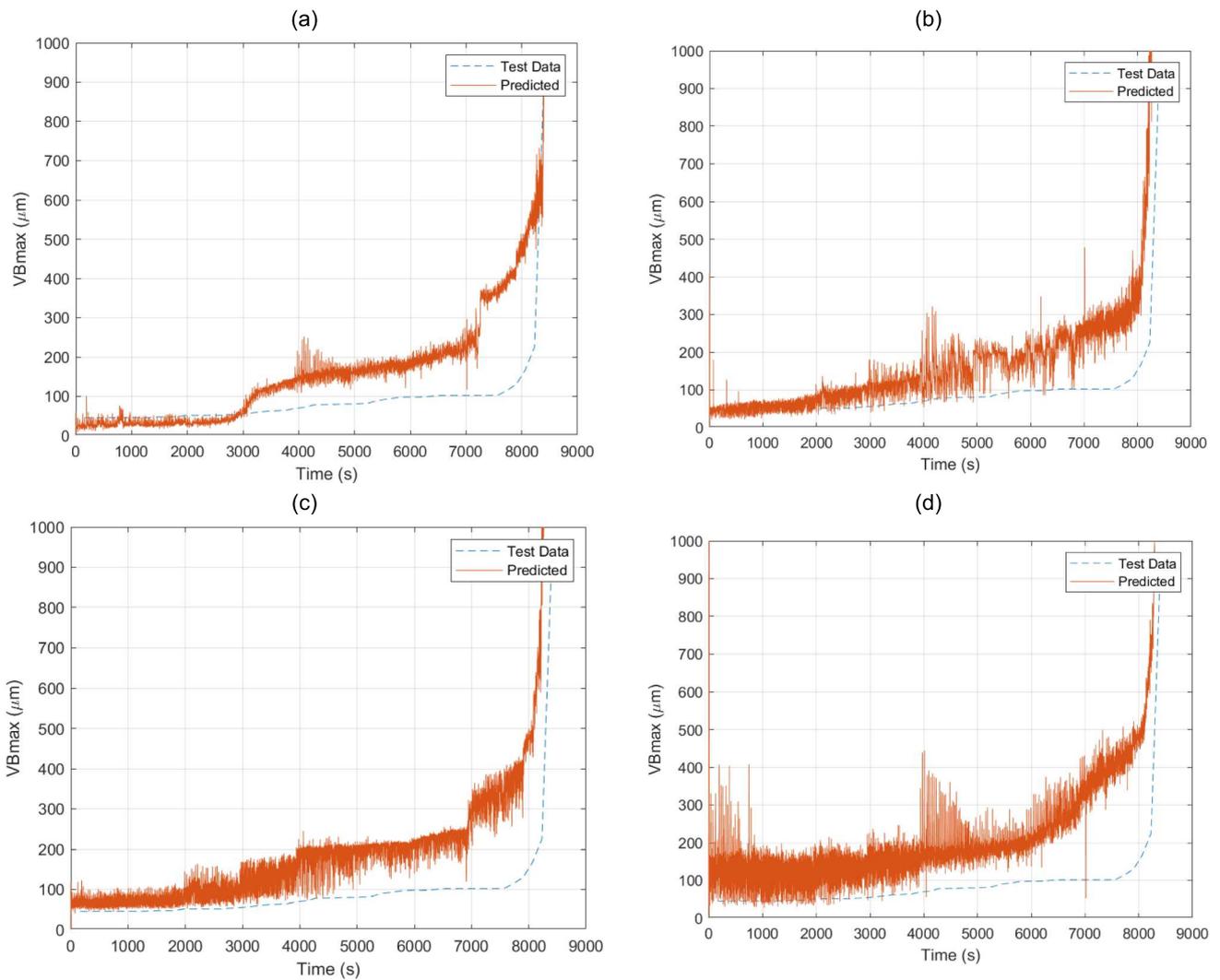


Fig. 21 Prediction results for the top four networks on the real-time machining testing

Table 5 Prediction RMSE results for the top four networks on the real-time machining testing

Network No	Real-time testing RMSE (μm)
1	119.3621
2	145.3745
3	165.8503
4	171.4894

Conclusions

The current paper investigated the development and implementation of a digital twin framework for intelligent real-time tool wear prediction in a machining process. The key findings of this study are as follows:

- The deep learning models, optimised using Bayesian optimisation, demonstrated the ability to accurately predict

tool wear based on sensor data from the DOE machining tests with RMSE values as low as 33.17 μm .

- The real-time implementation of the digital twin framework on a new machining test validated its functionality in practical scenarios by yielding a prediction RMSE of 119.36 μm .
- The observed discrepancies between predicted and measured wear in the real-time testing highlighted the challenges associated with real-world sensor noise and potential variations in the machining process compared to

controlled environments. However, the predictions closely followed the overall wear trend, including capturing the inflection point of rapid wear increase.

- The novel digital twin framework demonstrates its potential for online tool wear monitoring with the ability to process live sensor data, generate real-time wear predictions, and update its internal state based on the latest information.
- Future work of this research should explore other DL models to enhance the digital twin framework's capabilities. T. Similarly, further data collection could also improve the generalisability of the DL models and their robustness to real-world variations. Additionally, incorporating a continuous wear monitoring element through an in-situ inspection system would enable tool wear verification. and continuous learning.

Acknowledgements The authors would like to thank Adam Brown for his guidance on the tool wear optical measurement system.

Funding This work was supported by HVM Catapult funding provided by Innovate UK (project numbers 160080 and 160109).

Declarations

Conflict of interest The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Ambhore, N., Kamble, D., Chinchankar, S., & Wayal, V. (2015). Tool condition monitoring system: A review. *Materials Today: Proceedings*, 2(4–5), 3419–3428. <https://doi.org/10.1016/J.MATPR.2015.07.317>
- ARTIS Marposs. (2022). MARPOSS tool and process monitoring system. Retrieved February 28, 2022, from <https://www.marposs.com/eng/product/tool-and-process-monitoring-system-2>
- Caron. (2022). TMAC: Tool monitoring adaptive control—Caron engineering. Retrieved February 28, 2022, from <https://www.caroneng.com/products/tmac>
- Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16, 321–357. <https://doi.org/10.1613/JAIR.953>
- Chen, Q., Xie, Q., Yuan, Q., Huang, H., & Li, Y. (2019). Research on a real-time monitoring method for the wear state of a tool based on a convolutional bidirectional LSTM model. *Symmetry (Basel)*. <https://doi.org/10.3390/SYM11101233>
- Colantonio, L., Equeter, L., Dehombreux, P., & Ducobu, F. (2021). A systematic literature review of cutting tool wear monitoring in turning by using artificial intelligence techniques. *Machines*. <https://doi.org/10.3390/MACHINES9120351>
- Dimla, D. E., & Lister, P. M. (2000). On-line metal cutting tool condition monitoring.: I: Force and vibration analyses. *International Journal of Machine Tools and Manufacture*, 40(5), 739–768. [https://doi.org/10.1016/S0890-6955\(99\)00084-X](https://doi.org/10.1016/S0890-6955(99)00084-X)
- Duan, J., Zhang, X., & Shi, T. (2023). A hybrid attention-based parallel deep learning model for tool wear prediction. *Expert Systems with Applications*, 211, 118548. <https://doi.org/10.1016/J.ESWA.2022.118548>
- Fernández-Delgado, M., Cernadas, E., Barro, S., Amorim, D., & Fernández-Delgado, A. (2014). Do we need hundreds of classifiers to solve real world classification problems? *The Journal of Machine Learning Research*, 15, 3133–3181.
- Frazier, P. I. (2018). A tutorial on Bayesian optimization. Retrieved February 25, 2025 from <https://arxiv.org/abs/1807.02811v1>
- Gao, Z., Liu, H., & Li, L. (2023). Data augmentation for time-series classification: An extensive empirical study and comprehensive survey. Preprint retrieved from <https://arxiv.org/abs/2310.10060>
- Guo, H., Lin, X., & Zhu, K. (2022). Pyramid LSTM network for tool condition monitoring. *IEEE Transactions on Instrumentation and Measurement*. <https://doi.org/10.1109/TIM.2022.3173278>
- Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735–1780. <https://doi.org/10.1162/NECO.1997.9.8.1735>
- ISO 3685. (1993). Tool-life testing with single-point turning tools. Retrieved August 14, 2024, from <https://www.iso.org/standard/9151.html>
- Jiang, Y., Drescher, B., & Yuan, G. (2023). A GAN-based multi-sensor data augmentation technique for CNC machine tool wear prediction. *IEEE Access*, 11, 95782–95795. <https://doi.org/10.1109/ACCESS.2023.3311269>
- Kumar, S., Kolekar, T., Kotecha, K., Patil, S., & Bongale, A. (2022). Performance evaluation for tool wear prediction based on bi-directional, encoder–decoder and hybrid long short-term memory models. *International Journal of Quality and Reliability Management*, 39(7), 1551–1576. <https://doi.org/10.1108/IJQRM-08-2021-0291/FULL/PDF>
- Kurada, S., & Bradley, C. (1997). A review of machine vision sensors for tool condition monitoring. *Computers in Industry*, 34(1), 55–72. [https://doi.org/10.1016/S0166-3615\(96\)00075-9](https://doi.org/10.1016/S0166-3615(96)00075-9)
- Li, X., Qin, X., Wu, J., Yang, J., & Huang, Z. (2022). Tool wear prediction based on convolutional bidirectional LSTM model with improved particle swarm optimization. *International Journal of Advanced Manufacturing Technology*, 123(11–12), 4025–4039. <https://doi.org/10.1007/S00170-022-10455-1/FIGURES/13>
- Liao, Z., Schoop, J. M., Saelzer, J., Bergmann, B., Priarone, P. C., Spletstößer, A., Bedekar, V. M., Zanger, F., & Kaynak, Y. (2024). Review of current best-practices in machinability evaluation and understanding for improving machining performance. *CIRP Journal of Manufacturing Science and Technology*, 50, 151–184. <https://doi.org/10.1016/J.CIRPJ.2024.02.008>
- Liu, X., Zhang, B., Li, X., Liu, S., Yue, C., & Liang, S. Y. (2023a). An approach for tool wear prediction using customized DenseNet and GRU integrated model based on multi-sensor feature fusion. *Journal of Intelligent Manufacturing*, 34(2), 885–902. <https://doi.org/10.1007/S10845-022-01954-9/FIGURES/17>

- Liu, Z., Lang, Z. Q., Zhu, Y. P., Gui, Y., Laalej, H., & Stammers, J. (2023b). Sensor data modeling and model frequency analysis for detecting cutting tool anomalies in machining. *IEEE Trans Syst Man Cybern Syst*, 53(5), 2641–2653. <https://doi.org/10.1109/TSMC.2022.3218536>
- Liu, Z., Lang, Z. Q., Gui, Y., Zhu, Y. P., Laalej, H., & Curtis, D. (2024). Vibration signal-based tool condition monitoring using regularized sensor data modeling and model frequency analysis. *IEEE Transactions on Instrumentation and Measurement*, 73, 1–13. <https://doi.org/10.1109/TIM.2023.3343825>
- Marani, M., Zeinali, M., Songmene, V., & Mechefske, C. K. (2021). Tool wear prediction in high-speed turning of a steel alloy using long short-term memory modelling. *Measurement*, 177, 109329. <https://doi.org/10.1016/J.MEASUREMENT.2021.109329>
- Martins, D. H. C. S. S., de Lima, A. A., Pinto, M. F., Hemerly, D. D. O., Prego, T. D. M., e Silva, F. L., Tarrataca, L., Monteiro, U. A., Gutiérrez, R. H., & Haddad, D. B. (2023). Hybrid data augmentation method for combined failure recognition in rotating machines. *Journal of Intelligent Manufacturing*, 34(4), 1795–1813. <https://doi.org/10.1007/S10845-021-01873-1/TABLES/11>
- MONTRONIX. (2022). SpectraNG. Retrieved February 28, 2022, from <https://www.montronix.com/en/products/new-spectrang.html>
- Moore, J., Stammers, J., & Dominguez-Caballero, J. (2020). The application of machine learning to sensor signals for machine tool and process health assessment. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*. <https://doi.org/10.1177/0954405420960892>
- Nordmann. (2017). Nordmann tool monitoring presentation. Hürth, Germany. Retrieved October 05, 2017, from <http://www.toolmonitoring.com/presentation.html>
- Petitjean, F., Ketterlin, A., & Gançarski, P. (2011). A global averaging method for dynamic time warping, with applications to clustering. *Pattern Recognition*, 44(3), 678–693. <https://doi.org/10.1016/J.PATCOG.2010.09.013>
- Sayyad, S., Kumar, S., Bongale, A., Kotecha, K., Selvachandran, G., & Suganthan, P. N. (2022). Tool wear prediction using long short-term memory variants and hybrid feature selection techniques. *International Journal of Advanced Manufacturing Technology*, 121(9–10), 6611–6633. <https://doi.org/10.1007/S00170-022-09784-Y/FIGURES/12>
- Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. *J Big Data*, 6(1), 1–48. <https://doi.org/10.1186/S40537-019-0197-0/FIGURES/33>
- Snoek, J., Rippel, O., Swersky, K., Kiros, R., Satish, N., Sundaram, N., Patwary, M., Prabhat, M., & Adams, R. (2015). Scalable Bayesian optimization using deep neural networks. In *32nd international conference on machine learning, ICML*. (vol. 3, pp. 2161–2170). <https://doi.org/10.48550/arxiv.1502.05700>
- Sun, I. C., Cheng, R. C., & Chen, K. S. (2022). Evaluation of transducer signature selections on machine learning performance in cutting tool wear prognosis. *International Journal of Advanced Manufacturing Technology*. <https://doi.org/10.1007/S00170-021-08526-W/TABLES/6>
- Wang, D., Liu, Q., Wu, D., & Wang, L. (2022). Meta domain generalization for smart manufacturing: Tool wear prediction with small data. *Journal of Manufacturing Systems*, 62, 441–449. <https://doi.org/10.1016/J.JMSY.2021.12.009>
- Ward, R., Sun, C., Dominguez-Caballero, J., Ojo, S., Ayvar-Soberanis, S., Curtis, D., & Ozturk, E. (2021). Machining Digital Twin using real-time model-based simulations and lookahead function for closed loop machining control. *International Journal of Advanced Manufacturing Technology*, 117(11–12), 3615–3629. <https://doi.org/10.1007/s00170-021-07867-w>
- Wu, J., Chen, X. Y., Zhang, H., Xiong, L. D., Lei, H., & Deng, S. H. (2019). Hyperparameter optimization for machine learning models based on Bayesian optimization. *Journal of Electronic Science and Technology*, 17(1), 26–40. <https://doi.org/10.11989/JEST.1674-862X.80904120>
- Yang, J., Wu, J., Li, X., & Qin, X. (2023). Tool wear prediction based on parallel dual-channel adaptive feature fusion. *International Journal of Advanced Manufacturing Technology*, 128(1–2), 145–165. <https://doi.org/10.1007/S00170-023-11832-0/TABLES/7>
- Zhang, X., Yu, T., Xu, P., & Zhao, J. (2022). In-process stochastic tool wear identification and its application to the improved cutting force modeling of micro milling. *Mechanical Systems and Signal Processing*, 164, 108233. <https://doi.org/10.1016/J.YMSSP.2021.108233>
- Zhou, Y., & Xue, W. (2018). Review of tool condition monitoring methods in milling processes. *International Journal of Advanced Manufacturing Technology*, 96(5–8), 2509–2523. <https://doi.org/10.1007/s00170-018-1768-5>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.