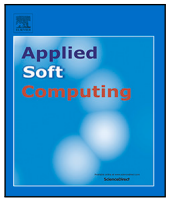
Applied Soft Computing Journal 148 (2023) 110922

Contents lists available at ScienceDirect

Applied Soft Computing

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



Bayesian-based uncertainty-aware tool-wear prediction model in end-milling process of titanium alloy

Gyeongho Kim [a], Sang Min Yang [b], Dong Min Kim [c], Sinwon Kim [b], Jae Gyeong Choi [a], Minjoo Ku [d], Sunghoon Lim [a,e,f,*], Hyung Wook Park [b,**]

[a] Department of Industrial Engineering, Ulsan National Institute of Science and Technology, 50, UNIST-gil, Ulsan 44919, Republic of Korea
[b] Department of Mechanical Engineering, Ulsan National Institute of Science and Technology, 50, UNIST-gil, Ulsan 44919, Republic of Korea
[c] Dongnam Division, Korea Institute of Industrial Technology, 25, Yeonkkot-ro 165beon-gil, Jinju 52845, Republic of Korea
[d] LG Electronics, 51, Gasan digital 1-ro, Seoul 08592, Republic of Korea
[e] Graduate School of Artificial Intelligence, Ulsan National Institute of Science and Technology, 50, UNIST-gil, Ulsan 44919, Republic of Korea
[f] Industry Intelligentization Institute, Ulsan National Institute of Science and Technology, 50, UNIST-gil, Ulsan 44919, Republic of Korea

ARTICLE INFO

Keywords:
Bayesian learning
Deep learning
Machining
Prognostics and health management
Titanium

ABSTRACT

Tool wear negatively affects machined surfaces and causes surface cracking, therefore increasing manufacturing costs and degrading product quality. Titanium alloys, which are widely used because of their desirable mechanical properties, have problems associated with tool wear due to poor thermal properties, such as specific heat capacity and thermal conductivity. Therefore, the accurate prediction of tool wear is necessary during the titanium alloy end-milling process to improve product quality and ensure reliability for corrective decisions like tool replacement. To this end, uncertainty-aware tool-wear prediction should be performed. In this study, a deep learning-based tool-wear prediction model based on a Bayesian approach was proposed. First, a convolutional neural network (CNN)-based architecture that integrates multiscale information extracted from raw sensor measurement data, termed deep multiscale CNN (DMSCNN), was proposed. It used different-sized convolutional kernels in parallel to enable various receptive field sizes suitable for machining processes. Second, based on a Bayesian learning approach, DMSCNN was transformed into a probabilistic model that produced a predictive distribution for estimated tool wear. In particular, a variational inference was applied to DMSCNN parameters to provide uncertainty awareness. Experiments were conducted with data collected from an actual end-milling process under three different conditions. The results proved the effectiveness of the proposed DMSCNN for tool-wear prediction. Bayesian DMSCNN showed promising results, as it outperformed existing comparative deterministic methods as well as probabilistic methods for tool-wear prediction. The proposed method is expected to be effectively applied in smart manufacturing as well as other machining processes that require data-driven digital decisions.

1. Introduction

Titanium alloy is an extraordinary material and is widely used in various industries, such as aerospace, medical, and automotive industries, owing to its desirable mechanical properties such as high strength-to-weight ratio, heat resistance, and corrosion resistance [1]. However, titanium alloy is classified as a difficult-to-cut material because of its low thermal conductivity and low specific heat, which lead to instability at elevated temperatures. High temperatures negatively influence tool wear and machined surface quality by facilitating chemical interactions and frictional volume losses at the tool–chip and tool–material interfaces [2]. In addition, poor mechanical properties cause surface cracks, while high residual stress reduces the fatigue life and increases the fatigue strength of the machined surface. These characteristics directly influence the useful life of a tool, which is

* Corresponding author at: Department of Industrial Engineering, Ulsan National Institute of Science and Technology, 50, UNIST-gil, Ulsan 44919, Republic of Korea
** Corresponding author at: Department of Mechanical Engineering, Ulsan National Institute of Science and Technology, 50, UNIST-gil, Ulsan 44919, Republic of Korea
E-mail addresses: kkh0608@unist.ac.kr (G. Kim), yangsangmin@unist.ac.kr (S.M. Yang), dkim0707@kitech.re.kr (D.M. Kim), swkim1123@unist.ac.kr (S. Kim), choil6043@unist.ac.kr (J.G. Choi), minjoo.ku@lge.com (M. Ku), sunghoonlim@unist.ac.kr (S. Lim), hwpark@unist.ac.kr (H.W. Park).

https://doi.org/10.1016/j.asoc.2023.110922
Received 30 January 2023; Received in revised form 27 August 2023; Accepted 6 October 2023
Available online 11 October 2023

strongly related to the machining cost (i.e., manufacturing cost). Therefore, the accurate prediction of the continuous wearing of cutting tools is required not only to prevent unexpected failures but also to improve productivity and quality by detecting tool breakage. Accurate prediction can also empower digital decisions and smart manufacturing like automating tool replacement decisions [3].

Despite the importance of tool-wear prediction for machining processes, the development of intelligent tool-wear prediction methods is a complicated task owing to multiphysics, chemical, and thermomechanical dynamics. However, several attempts have been made to develop tool-wear prediction methods based on prior knowledge in this field. The empirical formula for the degree of tool wear was studied based on the machining conditions, the geometry of the cutting tool, and material properties. In traditional approaches, tool wear is calculated based on certain cutting parameters by using a direct relationship. With the improved power of computerized computations, computer-aided engineering (CAE)-based methods have been developed, including the finite-element method (FEM), finite-difference method (FDM), and finite-volumetric method (FVM). In these methods, the work material and cutting tool are divided into several meshes, and the work material is defined using the Johnson–Cook plasticity model [4] with the appropriate material constant. In particular, specific cutting parameters, such as cutting force, strain rate, stress, and cutting temperature, were determined by analyzing chip formation and heat transfer [5]. As such, in FEM, the tool-wear-rate model is often used for the volumetric loss of the cutting-tool surface per unit time, and the derived tool-wear degree is used for further simulation processes.

However, traditional approaches to tool-wear prediction have significant limitations. Because the static information for the cutting parameters is only used to determine the degree of tool wear, the ongoing tool wear cannot be measured in real-time. Additionally, because the dynamically changing behavior of the equipment used for the machining processes is not included in the tool-wear estimation, traditional approaches cannot accurately predict the degree of tool-wear. To overcome these limitations, data-driven approaches that utilize online data collected with multiple sensors during machining processes have been used to predict tool wear and have become popular in recent years [6]. Data-driven approaches sensitively incorporate sudden changes in the machining process contained in the data. Therefore, it can better predict the ongoing tool wear adaptively, even under the same material, machining conditions, and stiffness of the tool and machine, whereas other conditions vary over time during the machining processes.

Among the data-driven approaches, those that can properly utilize multivariate time-series sensor measurement data for tool-wear prediction are the most sensible. In particular, prediction algorithms that can derive as much information as possible from complex raw data are required for accurate tool-wear prediction. Therefore, machine learning (ML)- and deep learning (DL)-based approaches have recently gained significant interest because they are capable of handling complex multivariate data with high expressive power. In particular, DL-based methods, which have shown superior performance in various domains, including computer vision, natural language processing, healthcare, and manufacturing [7–9], have been widely adopted. Therefore, this study proposes a DL-based tool-wear prediction method termed a deep multiscale convolutional neural network (DMSCNN). Considering the characteristics of the input data and actual experimental conditions, the proposed model is designed to be suitable for the end-milling process. By designing the specific architecture useful for analyzing multiscale features during machining operations, the proposed method is able to flexibly incorporate various contextual factors for tool-wear prediction.

Furthermore, this study proposes a novel uncertainty-aware tool-wear prediction model that has several advantages over the typical DL-based method. When uncertainty is incorporated during tool-wear prediction, each prediction result produced by the model is accompanied by the level of prediction confidence. This characteristic of uncertainty awareness could improve the prediction method's interpretability and reliability. To this end, this study adopts Bayesian learning, one of the most theoretically grounded and practical uncertainty estimation methods [10]. In particular, based on the proposed DMSCNN architecture, a Bayesian treatment was employed to enable the Bayesian approach for learning and inference, leading to a proposed method termed the Bayesian DMSCNN. Due to this uncertainty-aware wear prediction of high-cost tools, making decisions and corrective plans, such as when to replace tools based on the DL-based method, can become more robust.

The proposed data-driven tool-wear prediction method not only enables uncertainty awareness but also improves the prediction performance. Using sufficient experimental data from actual scenarios, the effectiveness of the proposed method was thoroughly validated. It can be further applied to diverse machining processes as well as decision-making scenarios in smart manufacturing, such as predictive maintenance and health management. The contributions of this study to the literature are as follows:

- A deep learning-based tool-wear prediction model (DMSCNN) is proposed, which is suitable for the end-milling process.
- Using Bayesian learning, the uncertainty-aware tool-wear prediction method is developed, termed the Bayesian DMSCNN.
- The effectiveness of the proposed method, including DMSCNN and Bayesian DMSCNN, is validated using experimental data collected from actual end-milling processes with different machining conditions.
- The proposed method shows superior performances compared to existing deterministic and probabilistic tool-wear prediction models.
- The efficacy of the proposed uncertainty-aware Bayesian DMSCNN in producing reliable confidence intervals and its desirable characteristics are discussed.

A preliminary version of this study was previously presented [11]. The remainder of this paper is organized as follows. Section 2 presents existing studies on tool-wear prediction and delineates preliminaries related to Bayesian learning. The proposed method, including a deep convolutional neural network (CNN)-based network architecture and its Bayesian treatment model as well as a theoretical basis, is detailed in Section 3. Section 4 provides a detailed illustration of the end-milling process setup, data collection, preprocessing, hyperparameter tuning, and implementation. In Section 5, experiments are conducted using deterministic tool-wear prediction models, including the proposed architecture, and their performances are compared. In addition, experimental results using the proposed Bayesian tool-wear prediction method and existing probabilistic models are provided. Thereafter, a post-hoc analysis of the probabilistic models is provided. Finally, in Section 6 the conclusions and future work are presented.

2. Preliminaries and literature review

This section discusses existing works in the literature related to tool-wear prediction, including traditional and recent data-driven approaches. Traditional approaches to tool-wear prediction based on physical models are reviewed, and their limitations are discussed. In addition, data-driven approaches, including those based on ML and DL, which improve not only the computational speed but also the precision of tool-wear prediction, are reviewed. In addition, Bayesian learning methods in ML literature are illustrated. Some existing Bayesian approaches to tool-wear prediction and their limitations are also discussed.

2.1. Traditional tool-wear prediction

The tool wear type is defined by its region on the cutting tool as well as the frictional mechanism. Flank wear occurs on the flank face and is most likely to occur owing to the abrasive and adhesive frictional mechanisms. Abrasive wear occurs when two bodies are in contact with each other by plowing a smoother surface; therefore, it is commonly generated at the flank face of the cutting tool and work material interface. Adhesive wear is micro-welded at the tool-workpiece interface. These two frictional mechanisms are particularly dominant in areas with low cutting speeds. However, crater wear occurs on the rake face because of diffusion and chemical wear. In particular, diffusion and chemical wear are dominant at elevated temperatures during metal cutting processes.

As mentioned above, the machining process has highly nonlinear multiphysics and chemical reactions that can be modeled for predicting tool wear. Therefore, several previous studies have been conducted to recognize the complexity of tool wear dynamics using mechanistic modeling [12]. Bjerke et al. presented a thermodynamic modeling framework for tool wear considering the interaction of chemical, oxidational, and diffusional reactions at the tool-chip interface [13]. Muñoz-Escalona et al. proposed an empirical tool-wear model based on thorough tool-wear analysis during the face-milling process [14]. Huang et al. proposed an analytical tool-wear model using various frictional coefficients, such as adhesion, abrasion, and diffusion coefficients [15]. Marksberry et al. developed a mechanistic tool-wear model under near-dry machining (NDM) conditions [16]. While the mechanistic modeling-based approaches can incorporate existing comprehensive domain knowledge for predicting tool wear, they are difficult to formulate to achieve a moderate level of accuracy. In addition, it is challenging to model realistic machining settings that may often vary in real-world scenarios.

On the other hand, finite-element analysis (FEA) simulation has also been widely used to predict tool wear owing to its complex interactions, realistic representations, and parameter sensitivity analyses. Moreover, FEA-based methods can generate failure mechanisms that lead to tool wear, therefore providing valuable insights for developing strategies to mitigate wear and enhance tool life. Yen et al. performed tool-wear estimation using FEA and geometric updates in an orthogonal cutting process [17]. Malakizadi et al. developed a three-dimensional FE simulation by predicting the temperature and tool wear of the cutting tool [18]. Attanasio et al. simulated tool wear during the drilling process using a FEM-based method [19]. Many studies in the literature, along with those mentioned above, suggest that while FEM-based analysis can overcome the limitations of visualization problems and experimental saving, its high computational cost and time need to be improved.

The traditional tool-wear prediction has been developed based on domain knowledge. This implies that considerable effort is required to build and adapt existing on-site traditional tool-wear prediction methods. Further, traditional approaches have several limitations. First, many empirical and analytical models have strong assumptions, simplifications, and approximations that do not fully represent actual scenarios. Considering that various factors that change over time continually affect the generation of tool wear, traditional approaches have limitations in real-world machining practices. Second, the physics-based model generally incurs a considerable cost and time to develop and adjust the appropriate constants in the empirical model [20]. Considering the aforementioned limitations, a data-driven approach to tool-wear prediction can provide more efficient and adaptable methods. In particular, the data-driven approach is not only applicable to any machining process where data can be collected but also requires much less domain knowledge. Further, dynamically changing the tool characteristics and related factors that affect the ongoing tool wear can be efficiently modeled. Therefore, this study proposed a data-driven tool-wear prediction method based on DL techniques.

2.2. Data-driven tool-wear prediction

Unlike traditional methods for tool-wear prediction, data-driven approaches use data obtained during the manufacturing process to estimate tool wear. Because data collected from multiple sensor types during the machining process are often high dimensional and multimodal [7,21,22], they carry comprehensive information on the status of the process. Therefore, it is imperative to use appropriate techniques to extract a meaningful representation of the features. Data-driven approaches aim to utilize rich input signals to capture latent features that help the accurate prediction of ongoing tool wear. Most ML-based tool-wear prediction methods in the literature use two-stage approaches: (1) feature extraction/selection and (2) tool-wear prediction.

Statistical dimensionality reduction algorithms, such as singular value decomposition (SVD) and principal component analysis (PCA), are often used for feature selection to estimate tool wear [23]. They are also employed to identify and extract the dominant features in time-series signals from milling machines [24]. In addition to the selected features, ML-based prediction models have been used to estimate the tool wear or classify its status. Zhou et al. identified the dominant features of acoustic emissions collected by sensors by using PCA [25]. Thereafter, the selected features were used to predict tool wear using an autoregressive model. Shi and Gindy used PCA and a least-squares support vector machine (SVM) to extract features from raw signals and estimate the degree of tool wear, respectively, [26]. However, some statistical methods for feature extraction and dimensionality reduction are not suitable for time-series signals and are not scalable to large-scale data due to their high computational complexity.

Other signal-processing and transformation techniques have also been used for feature extraction. Kothuru et al. suggested an SVM-based tool-wear-classification method using a fast Fourier transformation (FFT) as a feature extraction technique [27]. Zhu and Liu employed a hidden semi-Markov model (HSMM) with features extracted using wavelet packet decomposition (WPD) for online tool-wear monitoring [28]. Zhang et al. used a symmetrized dot pattern (SDP) to extract features from signals, and a multi-covariance Gaussian process regression (GPR) to predict tool wear [29].

The second phase of conventional ML-based tool-wear prediction approaches is rather straightforward, in the sense that ML algorithms are constructed to estimate tool wear using extracted features as input variables. Kong et al. presented a whale optimization-algorithm-based SVM model to predict tool wear [30]. In [31], multiple ML classifiers were used, such as SVM, random forest, and k-nearest neighbors, based on acoustic signals. Kong et al. used GPR for a real-time tool-wear-assessment technique [32]. Geramifard et al. used a hidden Markov model (HMM)-based tool-wear-monitoring framework for a computerized numerical control (CNC) milling machine [33]. Compared with the two-stage approaches mentioned above that utilize ML-based prediction algorithms, the method proposed in this study does not require any handcrafted feature extraction phases, as it is purely based on DL techniques. Therefore, the proposed method achieves a more succinct and effective tool-wear prediction performance during the end-milling process.

2.3. Deep learning-based tool-wear prediction

A DL-based tool-wear prediction that uses various types of DNN is another branch of data-driven approaches. Compared to traditional tool-wear prediction approaches and conventional data-driven approaches, DL-based methods that utilize more expressive models had shown superior performance in using complex sensor data. In particular, considering the recent advances in sensors (e.g., dynamometer) that generate high-frequency data with rich information, DL-based methods were considered more suitable for online data-driven tool-wear prediction [34]. In fact, owing to the rich model capacity and high performance of DL, various studies have proposed the application of

DL techniques to tool-wear prediction. Shi et al. developed a tool-wear-monitoring framework based on a multiple stacked sparse autoencoder (SSAE) that incorporates features extracted from multiple sensor signals [35]. He et al. also used SSAE for tool-wear prediction using temperature signals [36]. Duan et al. developed a DL-based tool-wear prediction method with explainability based on a principal component analysis network [37]. Cheng et al. suggested using DL algorithms, particularly for a multi-step wear prediction of cutting tools [38].

Some types of DNN that are effective for handling multivariate time-series data from sensor signals, such as recurrent neural networks (RNN), CNN, and their variants, are widely utilized. Considering that these models had shown superior performance in challenging tasks like natural language processing, they are considered compelling in time-series prediction tasks. Zhang et al. developed an RNN-based tool-wear prediction method that captures long-term temporal dependencies [39]. Wang et al. used long short-term memory (LSTM) with features extracted from machining signals based on a stacked autoencoder (SAE) to estimate tool wear under variable machining conditions [40]. Shah et al. also proposed using LSTM based on acoustic emission and vibration signals to monitor tool wear during the face-milling process [41]. Wang et al. developed a gated recurrent unit (GRU)-based tool-wear prediction method that captures local temporal patterns and long-term dependencies using enhanced feature-learning techniques [42]. There had been several studies that proposed hybrid approaches integrating multiple DL models for tool-wear prediction [43]. Despite the wide use of RNN-based methods, they have inherent limitations for an online prediction of tool wear due to the recurrent operations. Since sequential operations are required for RNNs, they cannot be parallelized and require longer computational time. Furthermore, when the input signal is lengthy, the performance of RNN-based methods showed to decrease [44]. Therefore, these limitations hinder an efficient online application of RNN-based tool-wear prediction methods in practice.

There are several studies that utilized CNNs for efficient tool-wear prediction using various types of features. Huang et al. proposed a deep CNN-based tool-wear prediction method based on multi-domain feature fusion [45]. Similarly, Huang et al. utilized multisensory signals with CNN to predict tool wear [46]. Xu et al. developed a CNN-based method using a channel attention mechanism to capture diverse feature fusion in the machining process [47]. Due to model flexibility and capacity, CNNs were widely used for predicting tool wear using various sensor signals. Yang et al. proposed a CNN-based hybrid approach that combines manual features with automatic features to perform tool-wear prediction considering both local features and global dependencies [48]. Xu et al. used CNNs with a dilated operation and residual connections for efficient feature extraction for tool-wear prediction in the tapping process [49]. Some works also utilized CNNs for image-based data, including direct monitoring images of machining processes and Fourier-transformed signals [50]. The method proposed in this study also employs a deep CNN-based model that uses multiple kernels of different sizes to achieve a larger receptive field, while being suitable for the end-milling process. As mentioned above, the proposed method utilizes the modeling flexibility of the CNN architecture that provides adjustable feature extraction capability. Another major contribution of this study differentiating it from previous studies that use DL-based approaches is the use of the Bayesian learning approach. Although DL-based approaches showed efficacy in predicting tool wear, they lack uncertainty awareness and therefore cannot represent prediction confidence. The proposed method addressed this inability with the Bayesian learning-based DL approach to improve prediction reliability and informed decision-making in practice.

2.4. Bayesian learning

Recently, the most widely used ML and DL algorithms have been trained based on the frequentist approach. The most widely used learning principle is the maximum likelihood estimation (MLE), in which the optimal set of parameters of a model is set to maximize the likelihood $p(D|\boldsymbol{w})$. For example, minimizing the squared error for the regression task and the cross-entropy for the classification task falls under the MLE principle. However, frequentist approaches have limitations because the trained parameters are deterministic point estimates. This induces predictive models to output a deterministic conditional average [51], and not to consider uncertainty in prediction. Furthermore, the unimodal nature of loss functions prohibits the estimation of variances [52]. These limitations make frequentist approaches susceptible to reliability problems.

Unlike MLE, the Bayesian learning (i.e., Bayesian inference) approach seeks to find a posterior distribution $p(\boldsymbol{w}|D)$ given data D that takes into account the uncertainty of the learning parameters. Based on Bayes' theorem (i.e., Bayes' rule), as shown in (1), $\boldsymbol{w} \in \mathbb{R}^d$, which is the set of parameters to be estimated, is updated using the given data.

$$p(\boldsymbol{w}|D) = \frac{p(D|\boldsymbol{w})p(\boldsymbol{w})}{p(D)} = \frac{p(D|\boldsymbol{w})p(\boldsymbol{w})}{\int p(D,\hat{\boldsymbol{w}})\,d\hat{\boldsymbol{w}}} = \frac{p(D|\boldsymbol{w})p(\boldsymbol{w})}{\int p(D|\hat{\boldsymbol{w}})p(\hat{\boldsymbol{w}})\,d\hat{\boldsymbol{w}}}. \tag{1}$$

One of the most attractive properties of Bayesian learning is inference (i.e., prediction), in which marginalization occurs as a process of inductive learning. During Bayesian inference, so called the Bayesian model average (BMA), a prediction of a new input x is made using the given likelihood function and a calculated posterior distribution. In particular, the likelihood $p(y|x,\boldsymbol{w})$ weighted by the posterior distribution is marginalized over model parameters $\boldsymbol{w}$ as shown in (2). This can also be interpreted as the use of all possible sets of parameters through marginalization (i.e., Bayesian ensemble), leading to the incorporation of epistemic uncertainty. In addition, this leads models to produce predictive distributions instead of deterministic point estimates, unlike the MLE approach.

$$p(y|x,D) = \int p(y|x,\boldsymbol{w})p(\boldsymbol{w}|D)\,d\boldsymbol{w}. \tag{2}$$

However, in most cases, the true posterior distribution (shown in (1)) is intractable or difficult to compute analytically; therefore, approximation methods are widely used. For example, sampling-based methods such as the Metropolis–Hastings algorithm, Gibbs sampling, and Markov Chain Monte Carlo (MCMC) have been widely used. In addition, considering the numerous model parameters, more computationally efficient approximation methods, such as stochastic gradient descent-based approximation, Laplace approximation, and expectation propagation, have been employed. In particular, for DNN, other approximate inference techniques have been proposed, including sampling-based, ensemble-based, and bootstrapping-based methods [53]. Variational inference (VI), which postulates an analytically simpler variational distribution as a substitute for the true posterior distribution, has been shown to be effective for DNN [54,55]. This study is also based on VI to endow the model parameters with uncertainty to provide uncertainty-aware tool-wear prediction. A detailed explanation of the proposed method related to the VI and its derivations is provided in Section 3.

2.5. Bayesian tool-wear prediction

Previous studies have applied Bayesian approaches to tool-wear prediction tasks. However, most of them have used Bayesian ML models that utilize Bayes' theorem at the algorithmic level, such as Bayesian networks and Bayesian classifiers. Karandikar et al. used a Naive Bayes classifier for tool-condition monitoring based on sensor measurements [56]. Shurrab et al. also suggested using Naive Bayes for the classification of cutting tool conditions in milling processes [57]. McParland et al. proposed a tool-wear prediction method for each force direction, based on a nonparametric Bayesian hierarchical Gaussian process [58]. Sun et al. developed a nonlinear Wiener process-based tool-wear prediction method on the basis of the Bayesian approach [59]. Li et al. utilized the Bayesian approach with various ML algorithms in a hybrid manner for tool condition monitoring [60].

However, these existing Bayesian approaches to tool-wear prediction are difficult to be applied to DL-based methods, which could limit prediction performance in practice. Therefore, it is necessary to develop a Bayesian-based tool-wear prediction method suitable for powerful DL algorithms.

Existing studies based on stochastic processes (i.e., state space models) have used the Bayesian inference concept. In particular, the Bayesian approach was widely employed in updating model parameters of HMMs and linear dynamical systems. Wang et al. proposed an enhanced particle filter (PF)-based tool-wear prediction method based on a Bayesian inference scheme [61]. Hanachi et al. also utilized a regularized PF to iteratively update the states to predict the tool wear [62]. Zhang et al. used a Bayesian-based updating scheme with particle learning for online tool-wear prognostics [63]. Wang and Gao proposed a Bayesian approach to predicting flank wear using the relation between tool wear and vibration signals [64]. Hao et al. applied a Bayesian update scheme to fuse real-time quality measurements with tool-wear rates in multistage manufacturing processes [65]. The aforementioned approach-based Bayesian tool-wear prediction methods use Bayesian inference only in part without uncertainty-aware characteristics. In addition, some limitations exist. For instance, Bayesian-based PF methods are computationally intensive, as they require a large number of particles. Furthermore, unlike the proposed method, they lack the scalability and efficiency required for large-scale online sensor signals for tool-wear prediction. To the best of our knowledge, there has been no existing work using the Bayesian approach with DNNs for tool-wear prediction. Considering this missing link between the Bayesian approach and the DL-based tool-wear prediction in the literature, as well as the necessity of uncertainty awareness, this study proposes a novel uncertainty-aware Bayesian-based method based on multiscale convolutions. The proposed method combines an expressive DL-based approach with suitable architecture and Bayesian learning to provide a novel tool-wear prediction method.

3. Proposed method

The tool-wear prediction method proposed in this study consists of two parts. First, a deep CNN-based model architecture that uses multiscale convolutional kernels is proposed. Second, using the proposed architecture, Bayesian learning was applied to transform the model into a probabilistic DL model that can produce uncertainty-aware predictions. In this section, not only are the architectural details of the model illustrated, but also the theoretical basis of the proposed method.

3.1. Proposed architecture

Considering the nature of the end-milling process, some concerns must be considered when developing a sensible tool-wear prediction model. First, compared to other domains where DL algorithms are used, such as natural language processing and computer vision, data collection is difficult in terms of time and cost. Furthermore, to collect labels to develop tool-wear prediction methods, experts in the field should manually measure the degree of tool wear, leading to increased annotation costs. Thus, because of the limited size of the data, the size of the model should also be limited [66]. At the same time, a model should have a proper receptive field size to pay attention not only to local temporal patterns but also to long-term dependencies between signals that are far apart from each other.

This study proposes a simple yet effective DNN architecture that considers the concerns mentioned above. First, the proposed architecture is purely based on CNN, which is considered more appropriate for lengthy sequence data, such as that used for tool-wear prediction. In fact, RNN and their variants (e.g., LSTM and GRU) have limitations in handling exceedingly long time-series data, as they recurrently update internal states, and the error should also be backpropagated through many time steps [67]. Although truncated backpropagation through time (BPTT) solves the issue in part, long-term dependencies are still not fully considered when using RNN-based algorithms with long sequences. However, CNN is free from this problem because it can vary the size of the receptive field with the manipulation of convolutional kernels to obtain multiscale features with different context information [68]. Furthermore, compared to RNN-based algorithms, CNN is faster owing to parallelization [44], making it faster in both training and inference.

Thus, this study presents a DMSCNN that uses different-sized convolutional kernels positioned in parallel. In particular, convolutional kernels with sizes of 3, 5, 7, and 9 are positioned in parallel within a single convolutional block to attain receptive fields of various sizes. As the data used to estimate tool wear in the end-milling process are multivariate time-series data, a one-dimensional (1D) convolution is used for all convolutional operations. Each input channel is assumed to be a single signal type (i.e., sensor type). Specifically, input to a single convolutional block first undergoes multiple convolutional operations with different-sized kernels. Then, each output of the parallel convolutional operation is concatenated to fuse various contextual information. Single convolutional operation of input f and kernel k with size l is calculated as (3). The calculation of a multiscale convolutional block using p parallel convolutional operations with different kernel sizes of l_p is expressed in (4).

$$f * k(i) = \sum_{j=1}^{l} k(j)\, f(i - j + \frac{1}{2}). \tag{3}$$

$$g = [g_1, \ldots, g_p] = [f * k^{l_1}, \ldots, f * k^{l_p}], \quad where\ |k^{s_i}| = l_i. \tag{4}$$

The proposed DMSCNN comprises multiple stacks of multiscale convolutional blocks with batch normalization [69], dropout [70], and an average pooling layer. After a series of multiscale convolutional blocks, a prediction module composed of global average pooling and fully connected layers is stacked to output the predicted tool wear. Using a global average pooling layer instead of a flattening layer has shown empirically better prediction performance. The overall architecture of the proposed DMSCNN is illustrated in Fig. 1. Detailed hyperparameter configurations and the tuning process are later illustrated in Section 4.

3.2. Bayesian learning using DMSCNN

The second part of the proposed method is based on the application of Bayesian learning using DMSCNN. As mentioned in Section 2, uncertainty-informed predictions can bring significant benefits in data-driven tool-wear prediction, especially when coupled with expressive DL-based methods. However, despite its practical effectiveness, uncertainty awareness is missing in the current literature on DNN-based tool-wear prediction. Among various uncertainty estimation methods, the Bayesian approach is deemed the most theoretically grounded and practical [55,71,72]. Bayesian learning can model uncertainty in prediction models through the posterior distribution computed given data during both training and inference (as shown in (2)). Therefore, this study applied a Bayesian learning approach to develop an uncertainty-aware tool-wear prediction model. There are some benefits to using the Bayesian approach to estimate tool wear. First, Bayesian inference enables a model to produce the prediction uncertainty represented in predictive distributions. In particular, compared with most DNN-based prediction methods that only provide a deterministic output, the Bayesian method provides distributions over the predictions. This not only provides a prediction confidence of the models but also enables domain experts in the field to selectively decide whether to use the prediction results [72]. In addition, using the Bayesian learning approach, improved prediction performance was achieved. In the following paragraphs, detailed explanations of the Bayesian treatment of DMSCNN and its theoretical basis are provided.

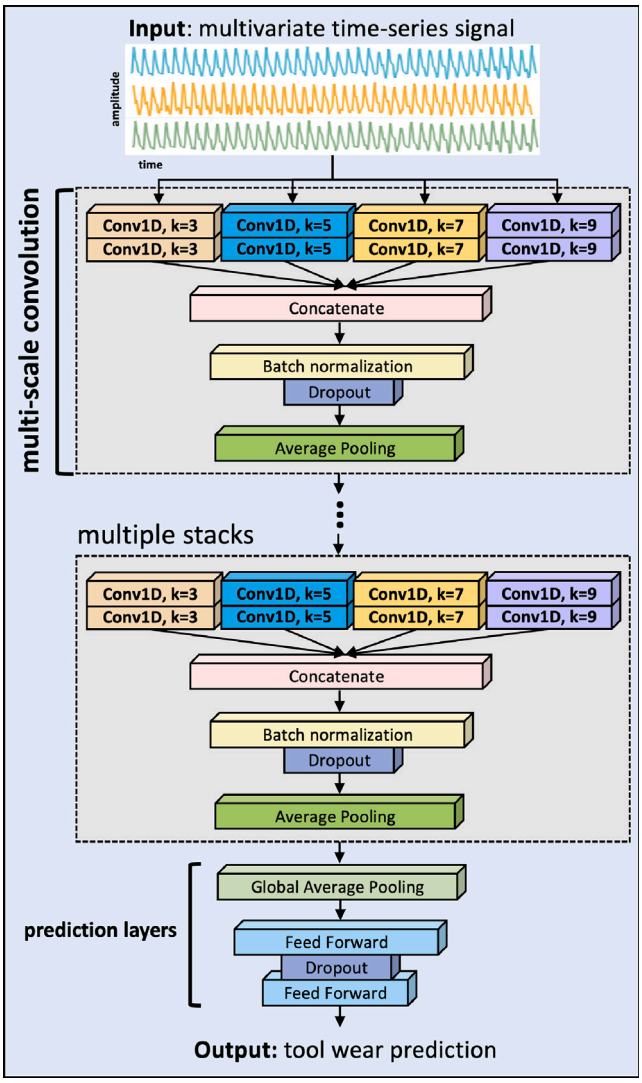


Fig. 1. Architecture of proposed deep multiscale convolutional neural network (DMSCNN).

First, a posterior distribution $p(\boldsymbol{w}|D)$ must be computed over the parameters of the DMSCNN. However, because a set of parameters $\boldsymbol{w}$ is high dimensional and the joint distribution is intractable, variational inference (VI) is employed, as [54,73]. As mentioned previously, VI is one of the most effective and efficient approximate inference approaches in Bayesian learning. Using VI, a variational distribution $q(\boldsymbol{w}|D)$, which is more tractable and easier to evaluate, is postulated to approximate the true posterior from which the DMSCNN parameters are assumed to be drawn. In this study, a mean-field VI that assumes a fully factorized variational distribution over parameters is used, as expressed in (5). This independence assumption has made the mean-field VI most widely used in Bayesian learning literature for its computational efficiency and scalability [74]. Recently, several alternative approaches to VI (e.g., generalized mean-field) that aim to estimate the covariance structure of parameters have been recently proposed. Nonetheless, considering the trade-off between scalability and approximation accuracy, mean-field VI is deemed most appropriate for practical applications. In addition, for large models, the mean-field VI was shown to better approximate the true posterior distribution [55].

$$p(\boldsymbol{w}|D) \approx q(\boldsymbol{w}|D) = \prod_{i=1}^{m} q(w_i|D) \tag{5}$$

To find an optimal variational distribution, the Kullback–Leibler divergence (KLD) between the true posterior $p(\boldsymbol{w}|D)$ and the variational posterior $q(\boldsymbol{w}|D)$ is minimized. Because KLD is always nonnegative, its minimization leads to a lower bound on KLD called the evidence lower bound (ELBO). The derivation of ELBO from the minimization of KLD is shown in (6). It leads to an optimization objective of ELBO expressed in (7). This implies that maximizing the ELBO can be interpreted as making variational distribution $q(\boldsymbol{w}|D)$ as close to prior $p(\boldsymbol{w})$ while considering the maximum likelihood objective $E_q[log\, p(D|\boldsymbol{w})]$. The ELBO can also be derived using a different approach, as shown in (8). By maximizing the ELBO through training, the variational distribution that is a surrogate for the true posterior during inference can be determined. This is identical to the optimization of the variational free energy (i.e., the Helmholtz free energy) or a minimum description length [75].

$$\begin{aligned}
KLD &= D_{KL}(q(\boldsymbol{w}|D)||p(\boldsymbol{w}|D)) \\
&= \int q(\boldsymbol{w}|D)\, log \frac{q(\boldsymbol{w}|D)}{p(\boldsymbol{w}|D)}\, d\boldsymbol{w} \\
&= \int q(\boldsymbol{w}|D)[log\, q(\boldsymbol{w}|D) - log\, p(\boldsymbol{w}|D)]\, d\boldsymbol{w} \\
&= -\int q(\boldsymbol{w}|D)[log \frac{p(D,\boldsymbol{w})}{p(D)} - log\, q(\boldsymbol{w}|D)]\, d\boldsymbol{w} \\
&= log\, p(D) - \int q(\boldsymbol{w}|D)[log \frac{p(D,\boldsymbol{w})}{q(\boldsymbol{w}|D)}]\, d\boldsymbol{w} \\
&= log\, p(D) - \int q(\boldsymbol{w}|D)[log \frac{p(\boldsymbol{w})p(D|\boldsymbol{w})}{q(\boldsymbol{w}|D)}]\, d\boldsymbol{w} \\
&= log\, p(D) - \int q(\boldsymbol{w}|D)[log \frac{p(\boldsymbol{w})}{q(\boldsymbol{w}|D)} + log\, p(D|\boldsymbol{w})]\, d\boldsymbol{w} \\
&= log\, p(D) - ELBO.
\end{aligned} \tag{6}$$

$$\mathcal{L}_{ELBO} = D_{KL}(q(\boldsymbol{w}|D)||p(\boldsymbol{w})) - E_q[log\, p(D|\boldsymbol{w})]. \tag{7}$$

$$\begin{aligned}
p(D) &\simeq log\, p(D) \\
&= log \int p(D,\boldsymbol{w})\, d\boldsymbol{w} \\
&= log \int p(D,\boldsymbol{w}) \cdot \frac{q(\boldsymbol{w}|D)}{q(\boldsymbol{w}|D)}\, d\boldsymbol{w} = log\, E_q[\frac{p(D,\boldsymbol{w})}{q(\boldsymbol{w}|D)}] \\
&\geq E_q[log \frac{p(D,\boldsymbol{w})}{q(\boldsymbol{w}|D)}] = \int q(\boldsymbol{w}|D)\, log \frac{p(D,\boldsymbol{w})}{q(\boldsymbol{w}|D)}\, d\boldsymbol{w} \\
&= \int q(\boldsymbol{w}|D)[log\, p(D,\boldsymbol{w}) - log\, q(\boldsymbol{w}|D)]\, d\boldsymbol{w} \\
&= ELBO.
\end{aligned} \tag{8}$$

In this study, a variational distribution is postulated as a Gaussian distribution, $\mathcal{N}(\mu, \rho^2)$ with parameters μ and ρ. In fact, there are other alternatives for variational distribution, such as Bernoulli, Laplace, von Mises-Fisher, and various mixture distributions [71]. Although these alternatives might provide richer expressivity for variational distributions, Gaussian distribution is most widely employed for several reasons. Most importantly, Gaussian-based VI achieves a desirable balance in trade-off between computational speed, tractability, and accuracy compared to the alternatives [76]. In addition, Gaussian distribution enables the analytical evaluation of KLD, which provides computational efficiency. Furthermore, Gaussian distribution helps to apply a reparameterization trick [73] for Bayesian DNNs, as it provides efficient gradient calculation. Therefore, although using other types of variational distributions might affect the prediction performance, the considerable advantages brought by using Gaussian variational distribution will be lost. For the aforementioned reasons, this model employs a Gaussian variational distribution with a reparameterization trick to ensure flawless backpropagation. Therefore, each parameter of DMSCNN w is drawn by sampling with Gaussian noise ϵ, as shown in (9). In other words, during the training of DMSCNN, optimal values for μ and ρ were sought rather than the exact model parameter w.

$$w = \mu + \rho \cdot \epsilon \quad where\ \epsilon \sim \mathcal{N}(0,1). \tag{9}$$

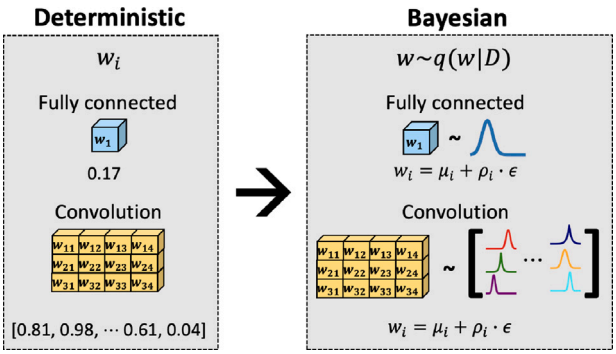


Fig. 2. Bayesian treatment of DMSCNN parameters.

Because the DMSCNN parameter w can now be drawn by sampling, the ELBO (7) is also approximated by Monte Carlo (MC) sampling, as shown in (10). ELBO optimization during training is trivial as both μ and ρ are linear with respect to w, hence any stochastic gradient algorithm can be employed. The prior distribution was set to Gaussian as $\mathcal{N}(0, 1)$. It is still worth noting that setting proper prior distributions over the network weights is an open challenge [77].

$$\tilde{\mathcal{L}}_{ELBO} = \frac{1}{n}\sum_{i=1}^{n}[log\ q(\boldsymbol{w}_{(i)}|D) - log\ p(\boldsymbol{w}_{(i)}) - log\ p(D|\boldsymbol{w}_{(i)})]. \tag{10}$$

Based on the aforementioned arguments, parameters of DMSCNN are converted into probability distributions that can be sampled during the inference. In the proposed Bayesian DMSCNN, all parameters, including those of convolutional layers and fully connected layers, were set under Bayesian treatment. A visualization of the Bayesian treatment on the parameters of DMSCNN is provided in Fig. 2.

After training, the proposed Bayesian DMSCNN was assumed to be uncertainty-aware. The uncertainty of the prediction of the model using the proposed method is represented in the sample variance of multiple stochastic passes during inference. Given a new input x whose tool-wear degree y is to be estimated, the predictive distribution can be obtained using (11). Each set of DMSCNN parameters $\boldsymbol{w}$ was drawn from $q(\boldsymbol{w}|D)$.

$$p(y|x, D) = \int p(y|x, \boldsymbol{w})p(\boldsymbol{w}|D)\, d\boldsymbol{w} \approx \frac{1}{T}\sum_{t=1}^{T} p(y|x, \boldsymbol{w}_{(t)}). \tag{11}$$

4. Experiments

4.1. Data collection

The end-milling process in this study was conducted using a conventional cutting fluid. Through a straight machining path, the material was machined with a milling type of down milling. A diameter of the cutting tool with 16 mm was used, and specific parameters of the cutting tool (F1200-8024775, Walter Corp.) are shown in Table 1. The z-axis length of the cutting tool was calibrated, and the origin was aligned with the surface of the work material. For the end-milling processes, Ti-6Al-4V was used as the work material, whose width, depth, and height were 100 mm, 100 mm, and 100 mm, respectively. Moreover, the work material was composed of 89.93% of titanium, 6.02% of aluminum, and 3.85% of vanadium [78]. The five-axis CNC machine of the horizontal type (HTC-1000, HNK Corp.) was used. A dynamometer (9257B, Kistler Corp.) was used to acquire the cutting force data at a sampling rate of 20 kHz. The machining pass was defined as a machining length of 100 mm. To accurately measure the reliability of the performance of the proposed method, experiments were performed under three types of machining conditions. Each experiment had a different material removal rate (MRR), which affected the degree of tool wear. Therefore, different machining passes were set to identify

Table 1
Cutting tool parameters.

Parameter	Value
Cooling condition	Conventional cutting fluids
Tool diameter	16 mm
Hone radius	0.012 mm
Nose radius	0.5 mm
Clearance angle	1st 9°, 2nd 20°
Rake angle	13°
Helix angle	44°

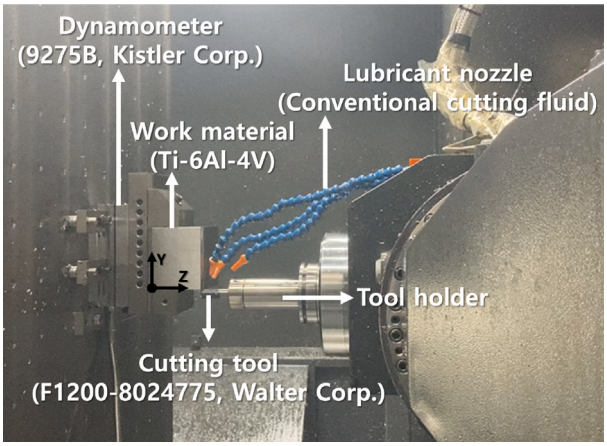


Fig. 3. The experimental setup for the end-milling process of Ti-6Al-4V.

obvious tool wear. The number of machining passes for the experiments was set to 94, 25, and 20 for Conditions 1, 2, and 3 respectively. The different machining conditions and machining lengths range from low to high MRR, as shown in Table 2. The experimental setup is shown in Fig. 3.

Three variables are used to predict tool wear: the forces on each axis, denoted as F_x, F_y, and F_z. The descriptive statistics for each set of experimental data are shown in Table 4, Table 3, and Table 5.

4.2. Tool-wear calculation

Tool wear was measured under all experimental conditions with specific machining distances. A digital microscope with laser confocal scanning capability (VHX-7000; Keyence Corp.) was used to measure tool flank wear with a magnification of 100x. Fig. 4 shows the measurement of the continuing tool wear with various machining distances.

Ideally, tool wear should be measured as frequently as possible during the machining process. However, this was not possible for several reasons. First, the removal and replacement of cutting tools for measurement of tool wear can cause a repetition of cooling and heating, leading to a surface crack in the cutting tool. Considering that the material properties of the cutting tool change rapidly during the machining process owing to the high temperature at the tool-workpiece interface, this is not feasible in an actual setting. Moreover, during cutting tool replacement, even minor adjustments to the cutting tool and its holder settings can potentially lead to alterations in the structural safety (e.g., tool stiffness) of the overall machining process.

Despite the limitations mentioned above, for tool wear to be predicted in real-time, continuous tool-wear data are essential for the development of a data-driven prediction model. To this end, this study uses a tool-wear equation based on Usui's tool-wear model [79]. The empirical tool-wear model was later extended by several researchers and is considered one of the most reliable tool-wear equations in the literature [80–82]. Several assumptions were required for using the tool-wear model; the variation of friction and wear mechanism is zero,

Table 2
Experiment conditions.

Experiment number	Cutting speed (mm/min)	Feed per tooth (mm/rev)	Axial depth (mm)	Radial depth (mm)	Machining distance (mm)	Material removal rate (mm^3/min)
1	60	0.08	5	1	9400	2860
2	100	0.1	5	1	2500	5970
3	120	0.1	5	1	2000	7160

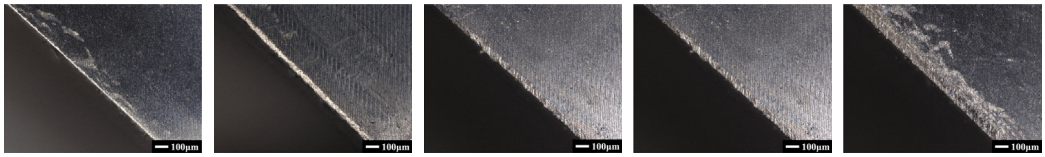


Fig. 4. Tool-wear measurement of Experiment 3 at (a) 4 passes, (b) 8 passes, (c) 12 passes, (d) 16 passes, and (e) 20 passes.

Table 3
Descriptive statistics of Dataset 1.

Variable	Mean	SD	Min	Max
F_x	141.4716	97.1282	−170.0590	518.4570
F_y	82.6874	80.5770	−190.9270	425.3610
F_z	−50.6475	46.8159	−248.7230	96.2546

Table 4
Descriptive statistics of Dataset 2.

Variable	Mean	SD	Min	Max
F_x	109.3085	125.6230	−504.0830	667.6810
F_y	110.7568	111.7301	−388.0790	533.6870
F_z	−59.3887	57.3354	−324.5350	127.1210

Table 5
Descriptive statistics of Dataset 3.

Variable	Mean	SD	Min	Max
F_x	225.7884	236.2957	−572.4660	944.2870
F_y	90.7225	139.4700	−258.7080	562.4460
F_z	−73.6629	88.2275	−459.3040	134.3950

and the chatter or tool vibration is neglected. While the cutting tool's geometry and dependence on empirical constants to express its material properties and wear mechanism remain limitations, Usui's tool-wear equation provides efficient and robust estimation. The tool-wear equation used in this study (shown in (12)) estimates the degree of tool wear between the machining distances at which tool wear is measured. In this study, the Levenberg–Marquardt (LM) method [83] was used to approximate the experimental tool-wear-measurement data. The LM algorithm aims to minimize the difference between experimental data and model predictions by adjusting the model parameters. It is imperative to use well-organized data to optimize the LM parameters. In other words, the quality and representativeness of experimental data must be ensured. To this end, this study uses the average tool-wear values from five time measurements to ensure data quality. In addition, every tool wear measurement was carried out at the same position to minimize the effects of outliers and measurement noise. Furthermore, a sonication process was conducted to remove the impurities on the cutting tool's surface. The equations used for the LM method are (13), (14), and (15). The estimated parameter values of the tool-wear equation for each dataset are listed in Table 6.

$$VB = d(a + bT^c)^{-1}. \tag{12}$$

$$\mathbf{p}_{k+1} = \mathbf{p}_k - (J_\mathbf{r}^T J_\mathbf{r} + \mu_k \, diag(J_\mathbf{r}^T J_\mathbf{r}))^{-1} J_\mathbf{r}^T \mathbf{r}(\mathbf{p}_k), \; k \geq 0 \tag{13}$$

Table 6
Estimated parameter values of the tool-wear equation.

Experiment number	a	b	c	d
1	−0.0397	0.2491	−0.3517	1.9717
2	0.0041	2.0200	−1.9975	1.5545
3	−0.2373	0.3837	−0.1391	2.2207

$$J_\mathbf{r}(\mathbf{p}) = \begin{bmatrix} \frac{\partial r_1(\mathbf{p})}{\partial p_1} & \cdots & \frac{\partial r_1(\mathbf{p})}{\partial p_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial r_n(\mathbf{p})}{\partial p_1} & \cdots & \frac{\partial r_n(\mathbf{p})}{\partial p_m} \end{bmatrix} \tag{14}$$

$$\mathbf{r}(\mathbf{p}) = \begin{bmatrix} r_1(\mathbf{p}) \\ r_2(\mathbf{p}) \\ \vdots \\ r_n(\mathbf{p}) \end{bmatrix} = \begin{bmatrix} y_1 - f(x_1, \mathbf{p}) \\ y_2 - f(x_2, \mathbf{p}) \\ \vdots \\ y_n - f(x_n, \mathbf{p}) \end{bmatrix} \tag{15}$$

4.3. Data preprocessing

All the data used in the experiments were centered and normalized using standardization, as shown in (16). For the experiments, all data were initially separated into a training and test set with ratios of 80% and 20%. To prevent information leakage (i.e., data snooping) and overfitting, the statistics used for standardization (i.e., $\bar{x}$, s) are calculated only using the holdout training set. Later, the initial training set is used in hyperparameter tuning.

$$x_{scaled} = \frac{x - \bar{x}}{s}, \tag{16}$$

where:

x: original independent variable
$\bar{x}$: mean
s: standard deviation
x_{scaled}: scaled independent variable

As mentioned above, the data used in this study collected from the end-milling experiments are thus long multivariate time-series consisting of synchronous sensor measurements. To change the raw data to a proper shape for use in prediction models, sliding-window (i.e., rolling window) data preprocessing was applied. With a stride of 1, all data were cropped into multiple sub-sequences with identical lengths (i.e., 132 time steps). This preprocessing step is required not only to reshape the data into a proper format but also to predict the wear of the tool in real-time during inference [84]. The sliding-window preprocessing is shown in Fig. 5.

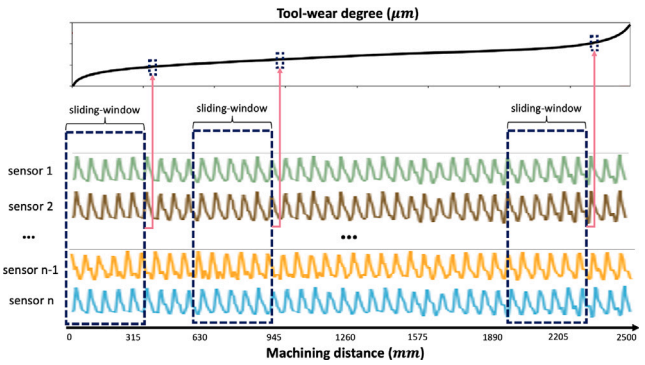


Fig. 5. Sliding-window preprocessing.

4.4. Evaluation metrics

As the target of prediction in this study was a continuous variable (i.e., degree of tool wear), evaluation metrics for regression tasks were used to assess the prediction performance. First, the mean absolute error (MAE) was used. In addition, the root-mean-squared error (RMSE), mean-absolute-percentage error (MAPE), and R^2 (coefficient of determination) were used. The evaluation metrics used were defined as follows:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i|. \tag{17}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}. \tag{18}$$

$$MAPE = \frac{100}{N}\sum_{i=1}^{N}\frac{|y_i - \hat{y}_i|}{|y_i|}. \tag{19}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{N}(y_i - \bar{y}_i)^2}, \tag{20}$$

where:

y: actual target value
$\hat{y}$: predicted target value
N: number of data

4.5. Hyperparameter tuning

The detailed architecture of the proposed DMSCNN used in the experiments is determined by various hyperparameters, including the number of convolutional blocks, the size of kernels, the number of kernels, the type of activation functions, etc. In addition, there are hyperparameters associated with training protocols, such as the batch size and the learning rate. The hyperparameters and the search space are provided in detail in Table 7. In this study, every hyperparameter of DMSCNN was tuned using a random search [85] based on a 10-fold cross-validation (CV), as shown in Fig. 6. First, the entire data was split into holdout training and test data. Then, using the training set, a 10-fold CV was performed, in which a hyperparameter search was conducted in each CV trial. The hyperparameter configuration that showed the highest performance on the CV validation fold was selected. During hyperparameter tuning, several hyperparameters, including the number of convolutional blocks, pooling strategy, and the type of activation functions, had shown to have greater effects on performance on the validation set than others.

Table 7
Hyperparameter tuning of DMSCNN.

Hyperparameter	Search space
Number of convolutional blocks	[2,10]
Number of kernels	{4,6,8,12,16,32,64}
Dropout rate	{0.1,0.2,0.3,0.4,0.5}
Pooling strategy	{average,max}
Pool size	{2,3}
Activation function	{Linear,tanh,ReLU,LeakyReLU}
Batch size	{16,32,64,128,256,512}
Learning rate	[0.0001,0.1]

4.6. Architecture details

Based on the hyperparameter tuning results, the obtained architectural details of DMSCNN are illustrated in Table 8. The unlisted values in the table are identical to the setting of the preceding convolutional block. For multiscale convolutional blocks, leaky ReLU was used as an activation function, and average pooling was employed. In terms of model complexity, the proposed DMSCNN has 138,753 trainable parameters, which is a moderate level of model complexity compared to existing conventional CNN architectures (e.g., VGG, ResNet) [86]. By employing the Bayesian learning approach, the proposed Bayesian DMSCNN has nearly twice as many trainable parameters as the naive DMSCNN. For computational complexity, DMSCNN requires 139.06 kFLOPs.

4.7. Implementation details

Using the optimal hyperparameter configuration obtained from the CV-based hyperparameter tuning, the training was performed on different CV trials as shown in Fig. 6. Since each CV trial has different CV folds, the model was trained on different training and validation sets for each time. Then, the evaluation was also performed independently 10 times to obtain the error bars (i.e., standard errors) for the results.

The experiments in this study were conducted using an Intel Xeon Gold 5220 CPU and an NVIDIA Tesla V100 GPU. For the implementation of DNN-based models, Python was used with TensorFlow (version 2.4.0) and TensorFlow Probability (version 0.16.0) [87]. During the training of the proposed method, negative log likelihood (NLL) was also used as a loss function. The Adam optimizer [88] was used as an optimization algorithm with a learning rate of 0.001 and a batch size of 256 based on the hyperparameter tuning results. Additionally, batch normalization [69] and early stopping with a patience of 100 epochs are used to prevent overfitting. During experiments, models are trained for a maximum of 50,000 epochs, where the best models are usually saved earlier. The training time for a single epoch took around 0.9 and 1.6 s on average for the DMSCNN and the Bayesian DMSCNN, respectively. During inference, the naive DMSCNN took 0.19 ms, whereas the Bayesian DMSCNN took 0.28 ms for each prediction.

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2, \tag{21}$$

where:

y: actual target value
$\hat{y}$: predicted target value
N: number of data

5. Results and discussion

In this section, the performance of the proposed tool-wear prediction method is validated in a two-stage setting. First, the effectiveness of the proposed DMSCNN in predicting the tool wear was verified. Thereafter, a Bayesian learning-based DMSCNN's performance is presented. In terms of training the two types of methods, the latter typically

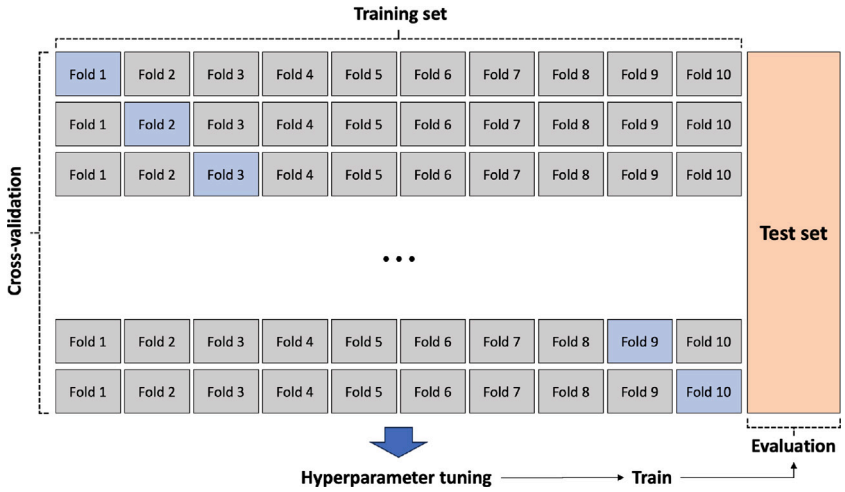


Fig. 6. Cross-validation performed in the experiments.

Table 8
The detailed architecture of the proposed DMSCNN.

Stack	Layer	Size of kernels	Number of kernels	Activation
MSConv block 1	Conv 1–1	3	8	LeakyReLU
	Conv 1–2	3	8	LeakyReLU
	Conv 2–1	5	8	LeakyReLU
	Conv 2–2	5	8	LeakyReLU
	Conv 3–1	7	8	LeakyReLU
	Conv 3–2	7	8	LeakyReLU
	Conv 4–1	9	8	LeakyReLU
	Conv 4–2	9	8	LeakyReLU
	Concatenate	/	/	/
	Batch norm	/	/	/
	Dropout (0.3)	/	/	/
	Average Pooling	2 (pool size)	/	/
MSConv block 2	–	–	–	–
MSConv block 3	Conv 1–1	3	16	LeakyReLU
	Conv 1–2	3	16	LeakyReLU
	Conv 2–1	5	16	LeakyReLU
	Conv 2–2	5	16	LeakyReLU
	Conv 3–1	7	16	LeakyReLU
	Conv 3–2	7	16	LeakyReLU
	Conv 4–1	9	16	LeakyReLU
	Conv 4–2	9	16	LeakyReLU
	Concatenate	/	/	/
	Batch norm	/	/	/
	Dropout (0.3)	/	/	/
	Average pooling	2 (pool size)	/	/
MSConv block 4	–	–	–	–
MSConv block 5	Conv 1–1	3	32	LeakyReLU
	Conv 1–2	3	32	LeakyReLU
	Conv 2–1	5	32	LeakyReLU
	Conv 2–2	5	32	LeakyReLU
	Conv 3–1	7	32	LeakyReLU
	Conv 3–2	7	32	LeakyReLU
	Conv 4–1	9	32	LeakyReLU
	Conv 4–2	9	32	LeakyReLU
	Concatenate	/	/	/
	Batch norm	/	/	/
Prediction module	Global average pooling	/	/	/
	FC 1	32	/	LeakyReLU
	FC 2	16	/	Linear
	Dropout (0.3)	/	/	/
	FC 3	8	/	Linear
	Output	1	/	Linear

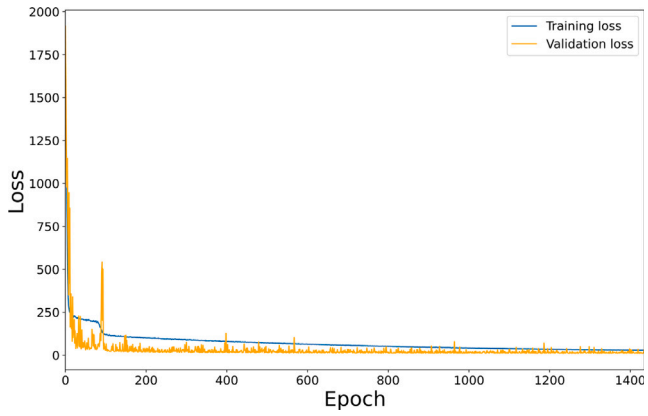


Fig. 7. Convergence analysis of DMSCNN training.

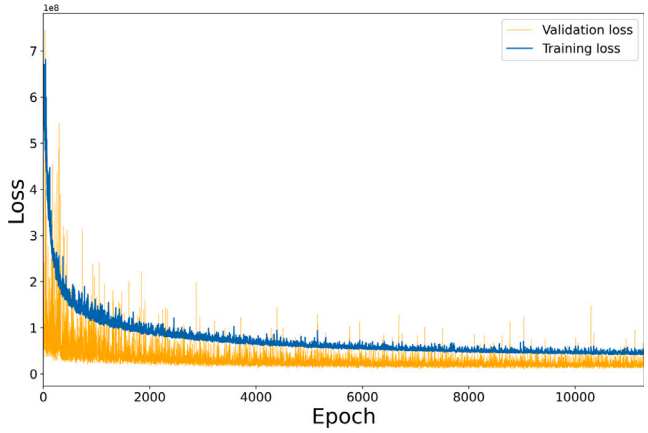


Fig. 8. Convergence analysis of Bayesian DMSCNN training.

requires a longer training time because it employs a Bayesian learning setting, which includes the sampling procedure and contains more parameters to estimate. The learning curves of the training and validation losses obtained from DMSCNN are shown in Fig. 7. In the case of the Bayesian DMSCNN, although the curves seem more fluctuating, both the training and validation losses decrease moderately, as shown in Fig. 8.

5.1. DMSCNN and deterministic tool-wear prediction

In this study, three types of data collected from different experimental setups were used, as mentioned in Section 4. First, the tool-wear prediction performance of the proposed DMSCNN was evaluated using the datasets. In particular, for performance comparison, other tool-wear prediction models that produce deterministic outputs were used. Comparative models include ML- and DL-based-prediction models that can handle time-series inputs, such as time-series forest (TSF) [89], GRU [90], LSTM [91], and 1D-CNN. In particular, 1D-CNN models with various kernel sizes (3, 5, 7, and 9) were used for comparison.

The prediction performances of the deterministic tool-wear prediction models, including the proposed DMSCNN on each different dataset used in this study, are provided in Table 9, Table 10, and Table 11, respectively. First, for Dataset 1, most DL-based prediction models, except TSF, showed similar performances in terms of the four evaluation metrics, as depicted in Table 9. However, the performance of the proposed DMSCNN was superior to that of all other prediction models by a large margin. For Dataset 2, as shown in Table 10, RNN-based prediction models, such as GRU and LSTM, exhibit better performance than 1D-CNN models. Moreover, the proposed DMSCNN showed higher performance than the other models for all evaluation metrics. For Dataset 3, the proposed DMSCNN confirmed its efficacy for tool-wear prediction, showing outperforming scores, as shown in Table 11.

Table 9
Performance comparison of deterministic tool-wear prediction models on Dataset 1.

Model	Metric			
	MAE	RMSE	MAPE	R^2
TSF	7.7813 (± 0.0101)	11.7798 (± 0.0111)	0.3043 (± 0.0016)	0.5255 (± 0.0008)
GRU	3.6248 (± 0.7324)	4.6265 (± 0.8914)	0.1077 (± 0.0189)	0.9229 (± 0.0279)
LSTM	3.9679 (± 0.2120)	5.0976 (± 0.2693)	0.1246 (± 0.0200)	0.9090 (± 0.0097)
1D-CNN (kernel_size = 3)	3.7913 (± 0.3624)	4.8126 (± 0.4432)	0.1219 (± 0.0146)	0.9182 (± 0.0146)
1D-CNN (kernel_size = 5)	3.6828 (± 0.7401)	4.6403 (± 0.8424)	0.1137 (± 0.0192)	0.9222 (± 0.0292)
1D-CNN (kernel_size = 7)	3.6379 (± 0.4657)	4.5613 (± 0.6206)	0.1121 (± 0.0166)	0.9257 (± 0.0196)
1D-CNN (kernel_size = 9)	3.4687 (± 0.6429)	4.2148 (± 0.7455)	0.1089 (± 0.0262)	0.9342 (± 0.0264)
DMSCNN (**proposed**)	**3.1394** (± 0.3892)	**3.9776** (± 0.4681)	**0.1002** (± 0.0091)	**0.9442** (± 0.0137)

Table 10
Performance comparison of deterministic tool-wear prediction models on Dataset 2.

Model	Metric			
	MAE	RMSE	MAPE	R^2
TSF	13.1047 (± 0.2720)	22.8500 (± 0.7443)	229.7708 (± 305.1545)	0.8513 (± 0.0100)
GRU	6.5810 (± 0.4025)	10.2392 (± 0.4280)	283.6626 (± 116.6763)	0.9698 (± 0.0025)
LSTM	6.5340 (± 0.7912)	10.1571 (± 1.0522)	500.1723 (± 318.6876)	0.9703 (± 0.0062)
1D-CNN (kernel_size = 3)	8.3762 (± 1.3030)	12.1755 (± 1.5131)	157.6921 (± 160.2621)	0.9572 (± 0.0105)
1D-CNN (kernel_size = 5)	7.0196 (± 1.2387)	10.3567 (± 1.4657)	124.3978 (± 170.0191)	0.9685 (± 0.0089)
1D-CNN (kernel_size = 7)	7.2237 (± 0.8353)	10.3698 (± 1.0951)	118.9289 (± 133.1617)	0.9682 (± 0.0057)
1D-CNN (kernel_size = 9)	6.8805 (± 1.0069)	9.9445 (± 0.9736)	72.3878 (± 33.8884)	0.9694 (± 0.0102)
DMSCNN (**proposed**)	**5.9349** (± 1.2047)	**8.8442** (± 1.5927)	**41.2545** (± 30.9068)	**0.9766** (± 0.0094)

Based on the observation that the proposed DMSCNN outperforms existing deterministic prediction methods in tool-wear prediction performance, the prediction results of DMSCNN on each dataset were visualized. The prediction results for the three datasets are presented in Figs. 9, 10, and 11, respectively. Compared to the ground-truth tool-wear degree denoted by the blue line, the prediction results of DMSCNN, shown as red points, indicate that DMSCNN can accurately predict the ongoing tool wear in the end-milling process. It is also observed that as the machining distance increases, the prediction error of DMSCNN also tends to show an increasing trend.

5.2. Bayesian DMSCNN and probabilistic tool-wear prediction

Given the previous results, the proposed DMSCNN exhibited superior performance over existing deterministic methods for tool-wear prediction. Subsequently, experiments were performed to verify the effectiveness of the DMSCNN with a Bayesian treatment, the so-called Bayesian DMSCNN. Because it provides probabilistic predictions, its performance is compared with existing probabilistic DL-based models,

Table 11
Performance comparison of deterministic tool-wear prediction models on Dataset 3.

Model	Metric			
	MAE	RMSE	MAPE	R^2
TSF	13.6732 (± 0.6169)	22.2633 (± 0.9599)	0.3116 (± 0.0186)	0.5881 (± 0.0392)
GRU	6.1292 (± 0.6965)	8.5532 (± 0.8459)	0.1288 (± 0.0202)	0.9359 (± 0.0108)
LSTM	7.1809 (± 0.8232)	10.0629 (± 0.9418)	0.1481 (± 0.0223)	0.9153 (± 0.0156)
1D-CNN (kernel_size = 3)	7.6203 (± 1.5108)	10.4687 (± 1.8086)	0.1509 (± 0.0309)	0.9027 (± 0.0339)
1D-CNN (kernel_size = 5)	7.7751 (± 1.6708)	10.4070 (± 1.6899)	0.1743 (± 0.0542)	0.9053 (± 0.0335)
1D-CNN (kernel_size = 7)	6.8475 (± 2.1101)	9.9393 (± 2.6305)	0.1372 (± 0.0416)	0.9100 (± 0.0505)
1D-CNN (kernel_size = 9)	7.5749 (± 1.6546)	13.2689 (± 8.5889)	0.1548 (± 0.0307)	0.9010 (± 0.0306)
DMSCNN (**proposed**)	**4.9355** (± 0.3681)	**6.9913** (± 0.5502)	**0.0956** (± 0.0164)	**0.9579** (± 0.0085)

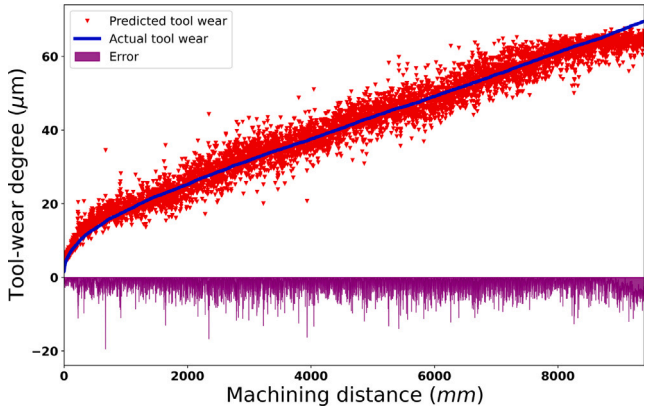


Fig. 9. Prediction results of DMSCNN on Dataset 1.

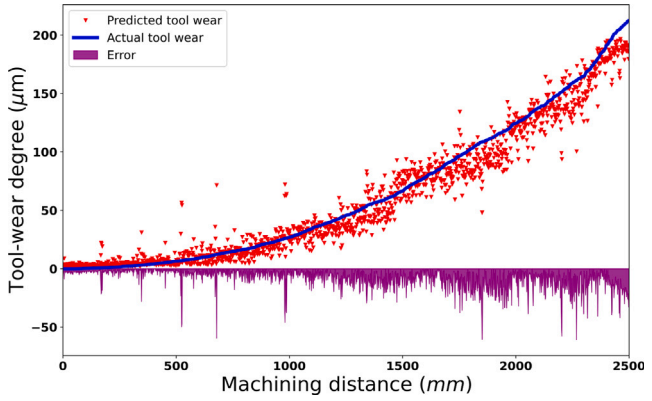


Fig. 10. Prediction results of DMSCNN on Dataset 2.

such as the Bayesian neural network (BNN), MC dropout-based LSTM, and CNN [10,54,71]. For the inference phase of all probabilistic models, including the Bayesian DMSCNN, MC estimation is used to produce the final prediction, as expressed in (2).

The tool-wear prediction performance of Bayesian DMSCNN and the comparative models on three datasets are provided in Table 12, Table 13, and Table 14, respectively. Although BNN is also able to consider predictive uncertainty via Bayesian learning, because it is not able to deal with time-series input data, it exhibited inferior performance to other methods in all datasets. MC dropout CNN exhibited better performance than MC dropout LSTM for all three datasets. Moreover, the proposed Bayesian DMSCNN outperformed other probabilistic DL-based prediction models in terms of all evaluation metrics.

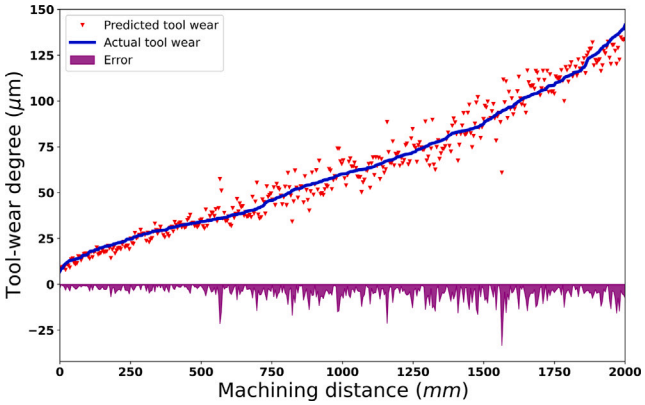


Fig. 11. Prediction results of DMSCNN on Dataset 3.

Table 12
Performance comparison of probabilistic tool-wear prediction models on Dataset 1.

Model	Metric			
	MAE	RMSE	MAPE	R^2
BNN	6.7235 (± 0.3380)	8.4677 (± 0.4005)	0.2113 (± 0.0131)	0.7482 (± 0.0235)
MC dropout LSTM	6.7012 (± 0.3741)	8.6475 (± 0.4824)	0.1748 (± 0.0067)	0.7390 (± 0.0274)
MC dropout CNN	4.7723 (± 0.1949)	6.0033 (± 0.2224)	0.1585 (± 0.0111)	0.8742 (± 0.0094)
Bayesian DMSCNN (**proposed**)	**2.3398** (± 0.1945)	**2.9862** (± 0.2430)	**0.0796** (± 0.0072)	**0.9737** (± 0.0069)

Table 13
Performance comparison of probabilistic tool-wear prediction models on Dataset 2.

Model	Metric			
	MAE	RMSE	MAPE	R^2
BNN	23.1017 (± 2.0104)	31.8812 (± 2.5667)	272.6718 (± 202.3328)	0.7139 (± 0.0476)
MC dropout LSTM	20.0668 (± 8.1623)	28.0024 (± 9.5189)	210.5684 (± 264.9961)	0.8225 (± 0.0369)
MC dropout CNN	12.2674 (± 1.3735)	17.3275 (± 2.2475)	250.3129 (± 219.4896)	0.9137 (± 0.0250)
Bayesian DMSCNN (**proposed**)	**5.3151** (± 0.5271)	**7.8672** (± 0.7653)	**29.8444** (± 32.9993)	**0.9839** (± 0.0037)

Table 14
Performance comparison of probabilistic tool-wear prediction models on Dataset 3.

Model	Metric			
	MAE	RMSE	MAPE	R^2
BNN	12.8891 (± 0.6742)	17.4075 (± 1.0464)	0.2540 (± 0.0227)	0.7431 (± 0.0341)
MC dropout LSTM	9.6906 (± 1.0792)	14.0705 (± 1.5207)	0.1680 (± 0.0165)	0.8367 (± 0.0328)
MC dropout CNN	7.6299 (± 0.6430)	10.4113 (± 0.6269)	0.1700 (± 0.0183)	0.9101 (± 0.0108)
Bayesian DMSCNN (**proposed**)	**4.1002** (± 0.3595)	**5.7841** (± 0.3797)	**0.0890** (± 0.0194)	**0.9731** (± 0.0043)

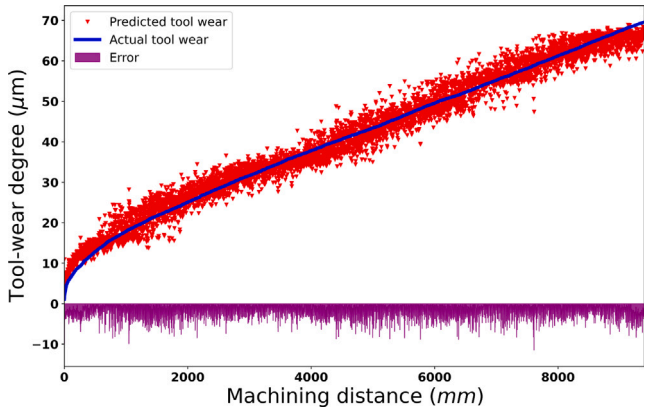


Fig. 12. Prediction results of proposed Bayesian DMSCNN on Dataset 1.

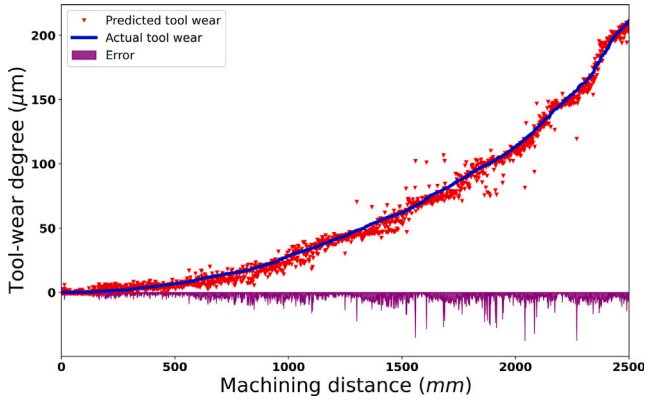


Fig. 13. Prediction results of proposed Bayesian DMSCNN on Dataset 2.

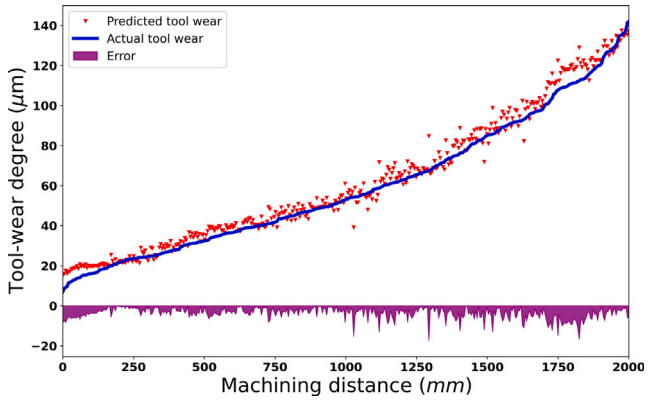


Fig. 14. Prediction results of proposed Bayesian DMSCNN on Dataset 3.

The prediction results of the proposed Bayesian DMSCNN for each dataset are shown in Figs. 12, 13, and 14. As the machining distance increased, the Bayesian DMSCNN continued to accurately predict tool wear, showing a trend similar to that of the naïve DMSCNN, as discussed previously. Furthermore, compared to DMSCNN, Bayesian DMSCNN not only shows a relatively smaller prediction error throughout every range of machining distance, but also exhibits a more consistent magnitude of the prediction error as the milling process proceeds. The results indicate that the proposed Bayesian DMSCNN is capable of accurately predicting the ongoing tool wear in various end-milling setups.

Compared to deterministic tool-wear prediction models, including naïve DMSCNN, probabilistic models can produce uncertainty-aware predictions. In particular, a predictive distribution can be obtained based on the probabilistic inference procedure using the constructed models. As discussed previously, when reasonable predictive uncertainty is derived from probabilistic models, it can be used effectively for digital decision-making scenarios, especially in risk-intolerant applications. For the three datasets, the predictive distributions generated using the comparative probabilistic models (i.e., BNN, MC dropout LSTM, and MC dropout CNN) and the proposed Bayesian DMSCNN are provided in Figs. 15, 16, and 17, respectively. The 95% confidence interval for each prediction output is shown with the mean prediction value and ± two standard deviations. For all three datasets, the predictive distributions of BNN exhibited unstable results, with inaccurate mean predictions and unreliable confidence intervals. For MC dropout LSTM, although the generated confidence intervals seem more reasonable at a glance, the range of confidence intervals is extremely large, indicating that the uncertainty is not properly captured. MC dropout CNN shows similar behavior of confidence intervals for datasets 1 and 2 as MC dropout LSTM; however, it shows more reasonable confidence intervals for Dataset 3. Finally, the proposed Bayesian DMSCNN exhibited the most reasonable range of confidence intervals for all three datasets. In most prediction cases, the Bayesian DMSCNN not only predicted the actual tool-wear degree well, but also provided moderate confidence intervals that were reliable. This result implies that the proposed method can be practical for end-users (e.g., domain experts) when performing tool replacement or maintenance, as it provides uncertainty estimates that provide the reliability of predictions. In practice, domain experts can utilize the predictions of the proposed method in controlling the conservativeness of tool replacement decisions.

Table 15
Inference time comparison of probabilistic models.

Model	Inference time (ms)
BNN	0.1632 ± 0.2237
MC dropout LSTM	3.4933 ± 0.1807
MC dropout CNN	0.2855 ± 0.4220
Bayesian DMSCNN (**proposed**)	0.2875 ± 0.2336

5.3. Post-hoc analysis of the proposed Bayesian DMSCNN

Using three datasets obtained from an actual end-milling process of titanium, the effectiveness of the proposed DMSCNN was validated. In addition, using a Bayesian treatment of DMSCNN, Bayesian DMSCNN produced outperforming tool-wear prediction results. Compared to the naïve DMSCNN, the proposed Bayesian DMSCNN has been shown to improve the prediction performance for the three datasets in terms of all evaluation metrics. Fig. 18 visualizes the improvement in tool-wear prediction performance by applying Bayesian treatment to DMSCNN in terms of MAE. In case of Dataset 2, the MAE has shown to be relatively high for both DMSCNN and Bayesian DMSCNN. This could be due to a rapid increase in tool-wear degree, leading to the vibration of the cutting tool [92].

In addition to comparing the prediction performance of the probabilistic models with that of the proposed Bayesian DMSCNN, which indicates the efficacy of the proposed method, the inference times of the probabilistic models were compared. However, the inference times of the deterministic models were not compared because they do not require a sampling procedure for MC estimation to yield predictive distributions. As shown in Table 15, the proposed Bayesian DMSCNN takes approximately 0.28 ms for each prediction, which is not only applicable to real-time prediction, but is also comparable to other probabilistic models. In particular, CNN-based models, including the Bayesian DMSCNN, show a much faster inference speed than the RNN-based model (i.e., MC dropout LSTM).

Finally, because the probabilistic models used in this study and the proposed Bayesian DMSCNN construct the variational distribution from

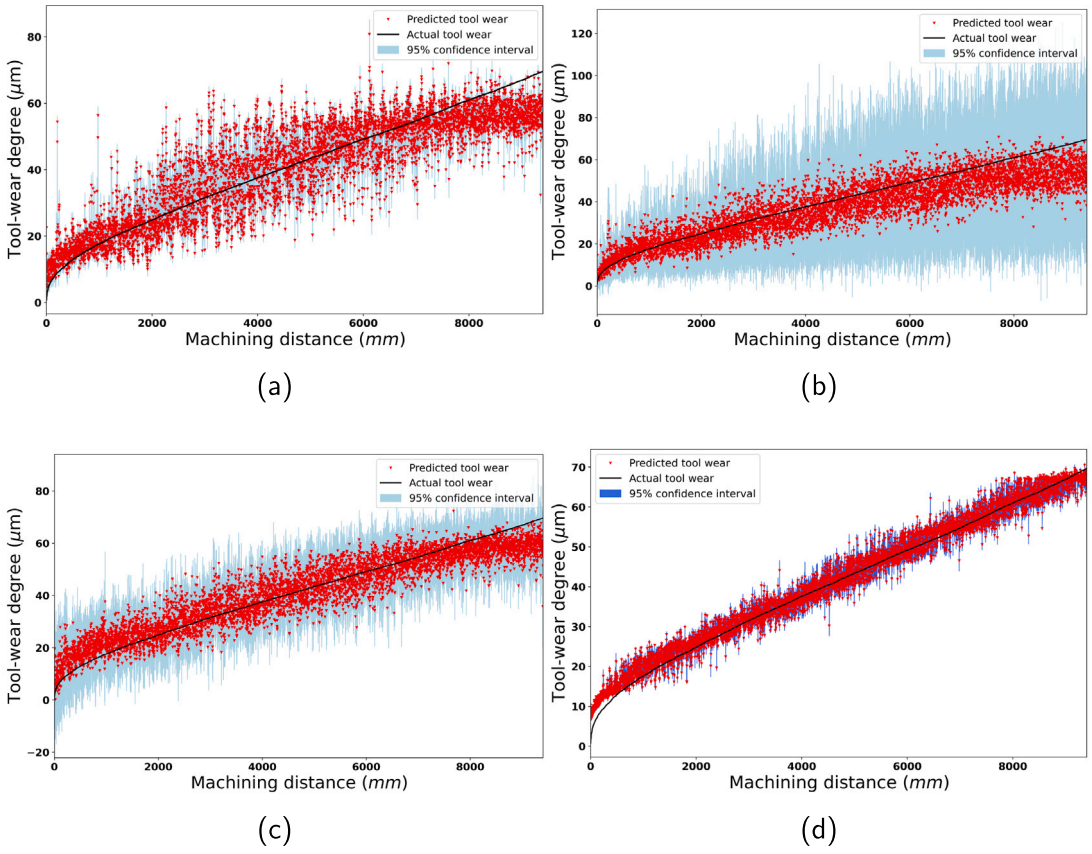


Fig. 15. Visualization of 95% prediction intervals of: (a) BNN, (b) MC dropout LSTM, (c) MC dropout CNN, and (d) proposed Bayesian DMSCNN on Dataset 1.

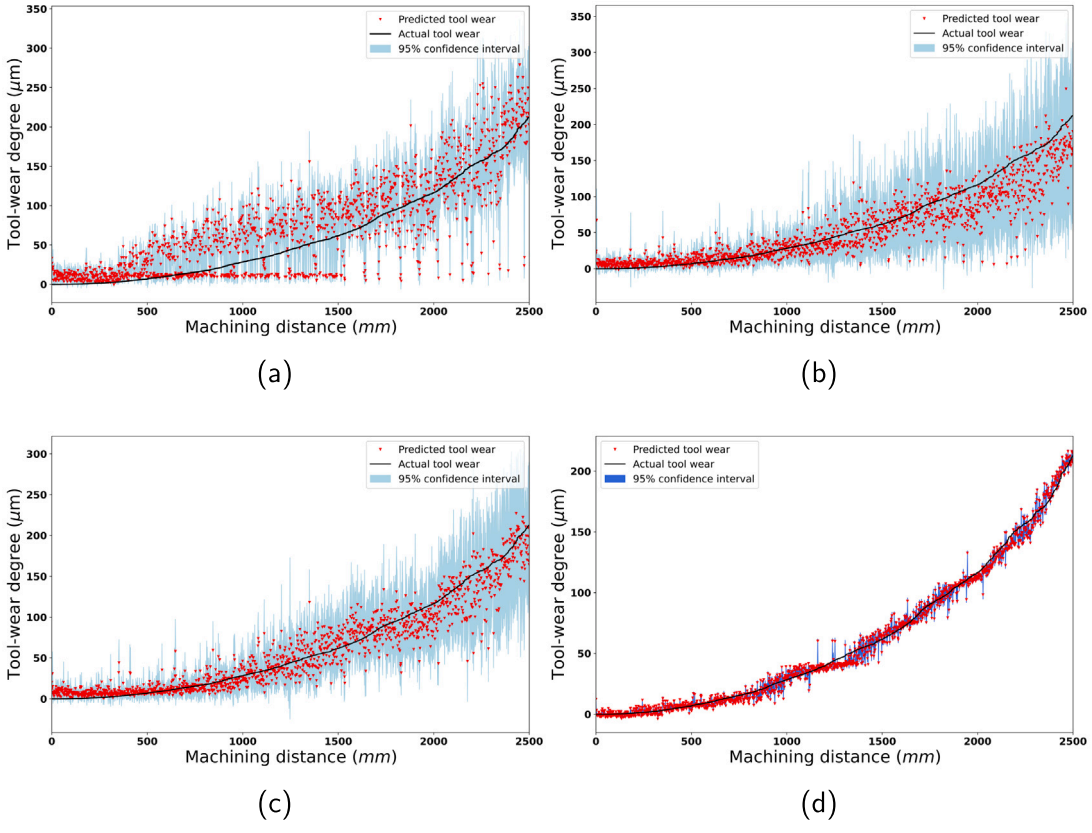


Fig. 16. Visualization of 95% prediction intervals of: (a) BNN, (b) MC dropout LSTM, (c) MC dropout CNN, and (d) proposed Bayesian DMSCNN on Dataset 2.

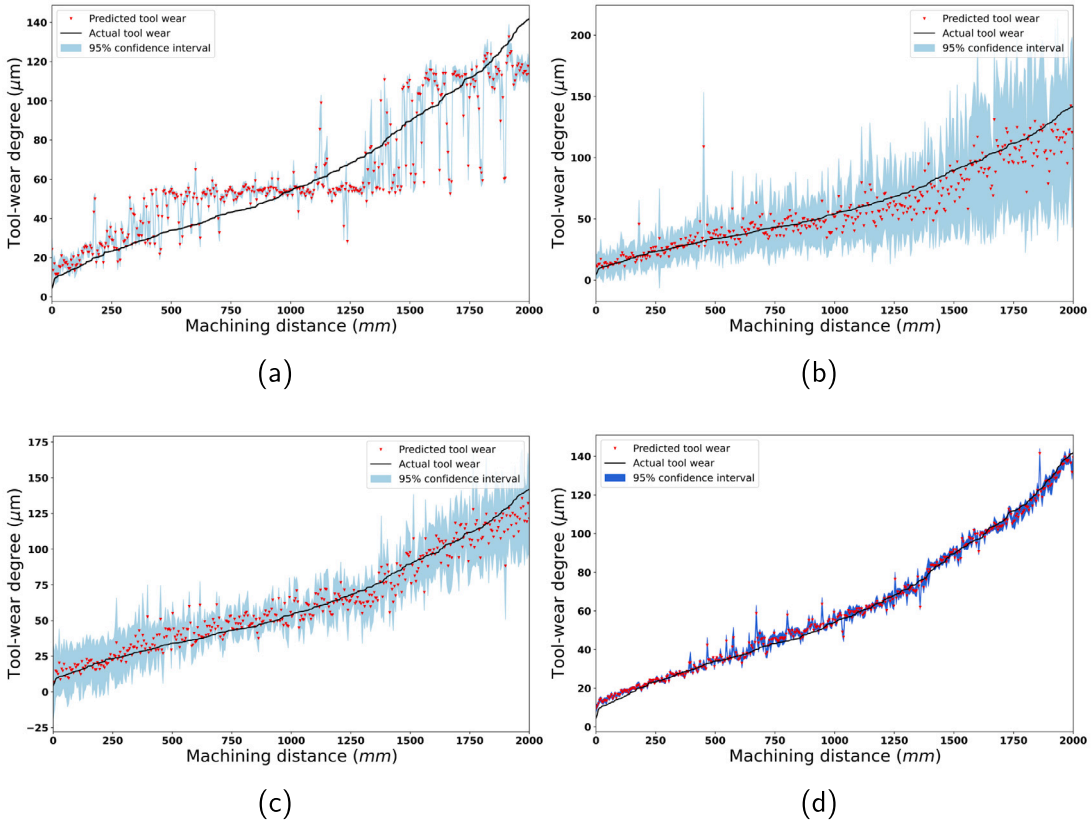


Fig. 17. Visualization of 95% prediction intervals of: (a) BNN, (b) MC dropout LSTM, (c) MC dropout CNN, and (d) proposed Bayesian DMSCNN on Dataset 3.

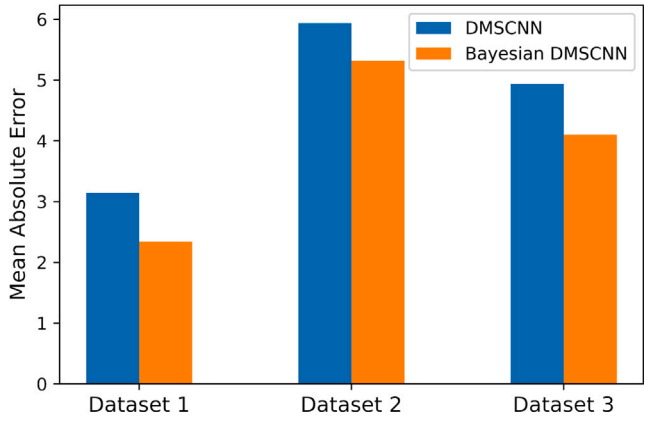


Fig. 18. Performance comparison of DMSCNN and Bayesian DMSCNN.

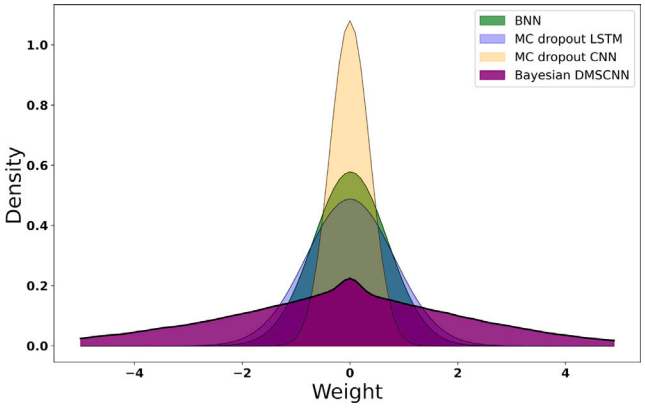


Fig. 19. Histogram of trained parameters from probabilistic models.

which the model parameters are sampled, the densities of the trained parameters are compared. Histograms of the trained parameters from the BNN, MC dropout LSTM, MC dropout CNN, and proposed Bayesian DMSCNN, as shown in Fig. 19. The trained parameter distribution of the proposed Bayesian DMSCNN was much wider than that of the comparative probabilistic tool-wear prediction models. This not only shows that the trained parameters are more diverse, but also provides a rationale for more reliable confidence intervals of the Bayesian DMSCNN.

6. Conclusion and future studies

This study addressed the problem of tool-wear prediction in the end-milling process of titanium using a data-driven approach based on DL and Bayesian learning. As tool-wear generated during the machining process of titanium alloy leads to detrimental effects on the overall process, accurate tool-wear prediction is imperative for improving the productivity and quality of machining products. Furthermore, because the cost of the tool itself and that required to replace machining tools are high, data-driven prediction results should be reliable for use in actual scenarios. To this end, this study proposes a Bayesian-learning-based tool-wear prediction method using a DL-based model that is suitable for the end-milling process. The effectiveness of the proposed DL-based model and its Bayesian-treated version were validated using data collected from three different end-milling processes.

The proposal for this study is twofold. First, based on the constraints of the tool-wear prediction problem in the end-milling process, an appropriate DL-based architecture termed multiscale CNN (DMSCNN)

was proposed. Using convolutional kernels of varied sizes that enable the adaptability of convolutional operations, the DMSCNN can accurately predict ongoing tool wear based on lengthy multivariate time-series input data. Second, based on the DMSCNN, a VI-based Bayesian learning technique was applied such that the prediction model could incorporate uncertainty for tool-wear predictions. The effectiveness of the proposed DMSCNN and Bayesian DMSCNN was validated using experimental data from actual applications. Compared with existing ML- and DL-based deterministic tool-wear prediction methods, the proposed DMSCNN outperformed conventional methods in all evaluation metrics (i.e., MAE, RMSE, MAPE, and R^2) in every dataset. In addition, the Bayesian DMSCNN showed superior tool-wear prediction performance compared with existing comparative probabilistic models for all datasets. The Bayesian DMSCNN has also shown more reliable and consistent confidence intervals for every dataset compared with comparative probabilistic models.

When comparing the naïve and Bayesian DMSCNN, the tool-wear prediction performance improved when Bayesian treatment was applied, with an additional ability to generate confidence intervals and predictive distributions. The Bayesian DMSCNN has also shown a sensible inference time while providing a greater diversity of trained model parameters, which further reinforces its verified efficacy observed in the experimental results.

The proposed method demonstrates the effectiveness of employing a Bayesian approach that considers uncertainty. In particular, the proposed method not only provided uncertainty awareness, but also improved the tool-wear prediction performance in the end-milling process. From a machining perspective, the range of effects on tool wear is wide; for instance, abrupt changes in the physical phenomenon (e.g., cutting force, temperature, vibration, etc.) at an early phase of machining can considerably affect the progress of tool wear at later phases. Thus, the Bayesian DMSCNN that can handle a wide range of multivariate time-series data and uncertainty during prediction is suitable for tool-wear prediction in the end-milling process. Additionally, with multiple stacks of multiscale convolutional operations and a variety of trained model parameters, the proposed Bayesian DMSCNN seems to have a higher expressive power. The reasonable uncertainty estimates, including predictive uncertainty obtained from the Bayesian DMSCNN, can be effectively used in various real-world applications. Making uncertainty-aware tool-wear predictions can help domain experts make informed decisions by taking into account reliability and potential risks. In addition, more accurate and reliable prediction results would lead to improved product quality, therefore enhancing customer satisfaction. From a more comprehensive perspective, for high-tech industries, where decision-making often accompanies considerable costs and time, uncertainty estimates can be useful in providing risk-informed decisions. For instance, considering not only the predicted values but also the uncertainty estimates can help cautious decision-making in practice where wrong decisions based on a single false prediction can bring significant loss. For other smart manufacturing domains where data-driven systems are used, uncertainty-aware predictions can also help make multifaceted decisions. Specifically, on-site domain experts can utilize predictive uncertainty as well as tool-wear prediction results obtained from the proposed Bayesian DMSCNN to control the conservativeness of the tool replacement.

The results of this study, which indicate the effectiveness of using multiscale features and uncertainty-aware prediction based on Bayesian learning, can be generalized to other areas as well. In fact, as the proposed method does not require prior domain knowledge about the machining dynamics and material characteristics for model training, it has high generalizability. In addition, the sensor signals inherently include the material properties and characteristics of the manufacturing process, since the sensors are constructed based on physical principles that govern how signals respond to changes in the properties of material and process. This enables the undemanding application of the proposed method that uses a solely data-driven approach. Therefore, when predicting using different work materials in the milling process, similar results are expected to be yielded for the proposed method when proper retraining and hyperparameter tuning are performed. This is also applied when the manufacturing process is changed because the predictive model can only be aligned with the changes in sensor signals that encapsulate material properties and process information. Furthermore, the results of this study can be generalized to related domains like other types of machining processes, such as subtractive manufacturing (e.g., turning, grinding, drilling) and additive manufacturing (i.e., 3D printing). Given data availability and proper training, the proposed method can be easily applied in the same data-driven fashion as this study [93]. When applying the proposed method to different scenarios, the degree and type of uncertainty should be taken into account, because some modifications to the model architecture might be required. In addition, input data specification, including the sampling rate and sequence length, should be taken into account for adopting the proposed method. This research has various implications in the field of applied soft computing. First, the development of data-driven soft computing techniques with suitable model architectures that consider practical settings is supported. In addition, this study exemplifies a thorough data-centric approach in terms of model design (i.e., DMSCNN) and operation (i.e., prediction) for applying soft computing techniques. Moreover, as empirical results validated the effectiveness and practical usefulness of uncertainty-aware predictions, this study promotes the application of uncertainty estimates in soft computing techniques to real-world problems. As the proposed method, employing uncertainty-aware DL-based models with Bayesian learning in other multidisciplinary soft computing tasks would also bring improved performances. Therefore, from an application perspective, this study implies that soft computing developments should have uncertainty awareness for the aforementioned reasons. In addition, this study highlights the importance of utilizing predictive uncertainty in risk-intolerant application domains to empower digital decisions with a richer data-driven basis. Similar approaches to this study can also have a greater impact by developing uncertainty-aware predictions for other machining prognostics and predictive maintenance tasks.

One of the limitations of the proposed method is that because it is based on Bayesian learning and VI, it requires higher computational complexity compared to the standard DNN with the same architectural form. Another limitation is that the ground-truth tool-wear degree was derived from the traditional tool-wear-calculation method. However, the proposed method is more applicable in the sense that multivariate time-series data collected during the end-milling process can be used to predict the degree of continuing tool wear online while providing uncertainty for each tool-wear prediction. Another limitation that should be considered, especially when applying the proposed method in other domains, is that the proposed method lacks the capability of multi-domain feature learning. Therefore, when machining conditions vary, learned features may have lower generalizability. Future studies will include the development of a sampling-free tool-wear prediction method with uncertainty awareness. Furthermore, the development of an interactive human–computer framework based on active learning for the intelligent prediction of tool wear remains a prospective extension of this study. Additionally, the application of the proposed method is planned in areas where uncertainty awareness is gaining importance, such as fault diagnosis and predictive maintenance [94].

CRediT authorship contribution statement

Gyeongho Kim: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Writing – original draft, Writing – review & editing, Visualization. **Sang Min Yang:** Conceptualization, Methodology, Formal analysis, Investigation, Writing – original draft, Writing – review & editing, Visualization. **Dong Min Kim:** Investigation, Writing – original draft, Project administration. **Sinwon Kim:** Conceptualization, Methodology, Writing – original draft, Writing –

review & editing. **Jae Gyeong Choi:** Validation, Formal analysis, Investigation, Writing – review & editing. **Minjoo Ku:** Validation, Formal analysis, Investigation, Writing – review & editing. **Sunghoon Lim:** Conceptualization, Methodology, Validation, Formal analysis, Investigation, Writing – original draft, Writing – review & editing, Visualization, Supervision, Project administration, Funding acquisition. **Hyung Wook Park:** Conceptualization, Methodology, Validation, Formal analysis, Investigation, Writing – original draft, Writing – review & editing, Visualization, Supervision, Project administration, Funding acquisition.

Declaration of competing interest

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

Data availability

Data will be made available on request.

Acknowledgments

This work was supported by the Advanced Technology Center Plus (ATC+) Program (20017932, 50% Accident Prevention Focus to reduce accident rate Development of Risk Detection System for Road Facilities Based on Artificial Intelligence) funded by the Ministry of Trade, Industry and Energy (MOTIE), Republic of Korea, the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2021R1F1A1046416), and Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. 2020-0-01336, Artificial Intelligence Graduate School Program (UNIST)). This work was supported by the Development of an on-site facility attached cryogenic machining integrated system funded by the Korea Institute of Industrial Technology (4/4) (KITECH EH-23-0011). This work was also supported by the Science and Technology Commercialization Promotion Agency grant funded by the Korea government in 2023 (Ministry of Science and ICT) (No. RS-2023-00254286). This work was also partly supported by AICP (AI Challengers Program) of UNIST (Ulsan National Institute of Science and Technology). (Gyeongho Kim and Sang Min Yang contributed equally to this work.)

References

[1] T. Narayanan, J. Kim, H. Park, High performance corrosion and wear resistant Ti-6Al-4V alloy by the hybrid treatment method, Appl. Surf. Sci. 504 (2020) 144388, http://dx.doi.org/10.1016/j.apsusc.2019.144388.

[2] S. Yang, J. Choe, J. Kim, H. Park, Improvement of tool life via unique surface modification of a tungsten carbide tool using a large pulsed electron beam in Ti-6Al-4V machining, J. Manuf. Process. 83 (2022) 223–234, http://dx.doi.org/10.1016/j.jmapro.2022.09.001.

[3] M. Rizal, J. Ghani, M. Nuawi, C. Haron, Online tool wear prediction system in the turning process using an adaptive neuro-fuzzy inference system, Appl. Soft Comput. 13 (2013) 1960–1968, http://dx.doi.org/10.1016/j.asoc.2012.11.043.

[4] G. Johnson, A constitutive model and data for materials subjected to large strains, high strain rates, and high temperatures, in: Proc. 7th Int. Symp. Ballistics, 1983, pp. 541–547.

[5] B. Li, A review of tool wear estimation using theoretical analysis and numerical simulation technologies, Int. J. Refractory Met. Hard Mater. 35 (2012) 143–151, http://dx.doi.org/10.1016/j.ijrmhm.2012.05.006.

[6] K. Patra, S. Pal, K. Bhattacharyya, Artificial neural network based prediction of drill flank wear from motor current signals, Appl. Soft Comput. 7 (2007) 929–935, http://dx.doi.org/10.1016/j.asoc.2006.06.001.

[7] G. Kim, J. Choi, M. Ku, H. Cho, S. Lim, A multimodal deep learning-based fault detection model for a plastic injection molding process, IEEE Access 9 (2021) 132455–132467, http://dx.doi.org/10.1109/ACCESS.2021.3115665.

[8] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (2015) 436–444, http://dx.doi.org/10.1038/nature14539.

[9] G. Kim, J. Choi, M. Ku, S. Lim, Developing a semi-supervised learning and ordinal classification framework for quality level prediction in manufacturing, Comput. Ind. Eng. 181 (2023) 109286, http://dx.doi.org/10.1016/j.cie.2023.109286.

[10] Y. Gal, Uncertainty in Deep Learning (Ph.D. dissertation), Dept. Eng. Univ. Cambridge, Cambridge, UK, 2016.

[11] G. Kim, S. Yang, S. Kim, D. Kim, S. Lim, H. Park, Tool wear prediction in the end milling process of Ti-6Al-4V using Bayesian learning, in: Proc. Int. Conf. Adv. Mechatronic Syst, 2022, pp. 64–69, http://dx.doi.org/10.1109/ICAMechS57222.2022.10003325.

[12] T. Wong, W. Kim, P. Kwon, Experimental support for a model-based prediction of tool wear, Wear 257 (2004) 790–798, http://dx.doi.org/10.1016/j.wear.2004.03.010.

[13] A. Bjerke, et al., Thermodynamic modeling framework for prediction of tool wear and tool protection phenomena in machining, Wear 484 (2021) 203991, http://dx.doi.org/10.1016/j.wear.2021.203991.

[14] P. Muñoz-Escalona, N. Díaz, Z. Cassier, Prediction of tool wear mechanisms in face milling AISI 1045 steel, J. Mater. Eng. Perform. 21 (2012) 797–808, http://dx.doi.org/10.1007/s11665-011-9964-6.

[15] Y. Huang, Y. Chou, S. Liang, CBN tool wear in hard turning: a survey on research progresses, Int. J. Adv. Manuf. Technol. 35 (2007) 443–453, http://dx.doi.org/10.1007/s00170-006-0737-6.

[16] P. Marksberry, I. Jawahir, A comprehensive tool-wear/tool-life performance model in the evaluation of NDM (near dry machining) for sustainable manufacturing, Int. J. Mach. Tools Manuf. 48 (2008) 878–886, http://dx.doi.org/10.1016/j.ijmachtools.2007.11.006.

[17] Y. Yen, J. Söhner, B. Lilly, T. Altan, Estimation of tool wear in orthogonal cutting using the finite element analysis, J. Mater Process. Technol. 146 (2004) 82–91, http://dx.doi.org/10.1016/S0924-0136(03)00847-1.

[18] A. Malakizadi, H. Gruber, I. Sadik, L. Nyborg, An FEM-based approach for tool wear estimation in machining, Wear 368–369 (2016) 10–24, http://dx.doi.org/10.1016/j.wear.2016.08.007.

[19] A. Attanasio, F. Faini, J. Outeiro, FEM simulation of tool wear in drilling, Procedia CIRP 58 (2017) 440–444, http://dx.doi.org/10.1016/j.procir.2017.03.249.

[20] J. Wang, Y. Li, R. Zhao, R. Gao, Physics guided neural network for machining tool wear prediction, J. Manuf. Syst. 57 (2020) 298–310, http://dx.doi.org/10.1016/j.jmsy.2020.09.005.

[21] C. Chien, C. Chen, Adaptive parametric yield enhancement via collinear multivariate analytics for semiconductor intelligent manufacturing, Appl. Soft Comput. 108 (2021) 107385, http://dx.doi.org/10.1016/j.asoc.2021.107385.

[22] M. Khakifirooz, C. Chien, Y. Chen, Bayesian inference for mining semiconductor manufacturing big data for yield enhancement and smart production to empower industry 4.0, Appl Soft Comput. 68 (2018) 990–999, http://dx.doi.org/10.1016/j.asoc.2017.11.034.

[23] W. Cheng, C. Cheng, Y. Lei, P. Tsai, Feature selection for predicting tool wear of machine tools, Int. J. Adv. Manuf. Technol. 111 (5) (2020) 1483–1501, http://dx.doi.org/10.1007/s00170-020-06129-5.

[24] J. Zhou, C. Pang, F. Lewis, Z. Zhong, Intelligent diagnosis and prognosis of tool wear using dominant feature identification, IEEE Trans. Ind. Inform. 5 (4) (2009) 454–464, http://dx.doi.org/10.1109/TII.2009.2023318.

[25] J. Zhou, C. Pang, Z. Zhong, F. Lewis, Tool wear monitoring using acoustic emissions by dominant-feature identification, IEEE Trans. Instrum. Meas. 60 (2) (2010) 547–559, http://dx.doi.org/10.1109/TIM.2010.2050974.

[26] D. Shi, N. Gindy, Tool wear predictive model based on least squares support vector machines, Mech. Syst. Signal Process. 21 (4) (2007) 1799–1814, http://dx.doi.org/10.1016/j.ymssp.2006.07.016.

[27] A. Kothuru, S. Nooka, R. Liu, Application of audible sound signals for tool wear monitoring using machine learning techniques in end milling, Int. J. Adv. Manuf. Technol. 95 (9) (2018) 3797–3808, http://dx.doi.org/10.1007/s00170-017-1460-1.

[28] K. Zhu, T. Liu, Online tool wear monitoring via hidden semi-Markov model with dependent durations, IEEE Trans. Ind. Inform. 14 (1) (2017) 69–78, http://dx.doi.org/10.1109/TII.2017.2723943.

[29] C. Zhang, W. Wang, H. Li, Tool wear prediction method based on symmetrized dot pattern and multi-covariance Gaussian process regression, Measurement 189 (2022) 110466, http://dx.doi.org/10.1016/j.measurement.2021.110466.

[30] D. Kong, et al., Tool wear estimation in end milling of titanium alloy using NPE and a novel WOA-SVM model, IEEE Trans. Instrum. Meas. 69 (7) (2019) 5219–5232, http://dx.doi.org/10.1109/TIM.2019.2952476.

[31] Z. Li, R. Liu, D. Wu, Data-driven smart manufacturing: Tool wear monitoring with audio signals and machine learning, J. Manuf. Process. 48 (2019) 66–76, http://dx.doi.org/10.1016/j.jmapro.2019.10.020.

[32] D. Kong, Y. Chen, N. Li, Gaussian process regression for tool wear prediction, Mech. Syst. Signal Process. 104 (2018) 556–574, http://dx.doi.org/10.1016/j.ymssp.2017.11.021.

[33] O. Geramifard, J. Xu, J. Zhou, X. Li, Multimodal hidden Markov model-based approach for tool wear monitoring, IEEE Trans. Ind. Electron. 61 (6) (2013) 2900–2911, http://dx.doi.org/10.1109/TIE.2013.2274422.

[34] Y. Chen, B. Zhou, M. Zhang, C. Chen, Using IoT technology for computer-integrated manufacturing systems in the semiconductor industry, Appl. Soft Comput. 89 (2020) 106065, http://dx.doi.org/10.1016/j.asoc.2020.106065.

[35] C. Shi, et al., Tool wear prediction via multidimensional stacked sparse autoencoders with feature fusion, IEEE Trans. Ind. Inform. 16 (8) (2019) 5150–5159, http://dx.doi.org/10.1109/TII.2019.2949355.

[36] Z. He, T. Shi, J. Xuan, T. Li, Research on tool wear prediction based on temperature signals and deep learning, Wear 478 (2021) 203902, http://dx.doi.org/10.1016/j.wear.2021.203902.

[37] J. Duan, C. Hu, X. Zhan, H. Zhou, G. Liao, T. Shi, MS-SSPCANet: A powerful deep learning framework for tool wear prediction, Robot. Comput.-Integr. Manuf. 78 (2022) 102391, http://dx.doi.org/10.1016/j.rcim.2022.102391.

[38] M. Cheng, et al., Intelligent tool wear monitoring and multi-step prediction based on deep learning model, J. Manuf. Syst. 62 (2022) 286–300, http://dx.doi.org/10.1016/j.jmsy.2021.12.002.

[39] J. Zhang, Y. Zeng, B. Starly, Recurrent neural networks with long term temporal dependencies in machine tool wear diagnosis and prognosis, SN Appl. Sci. 3 (2021) 1–13, http://dx.doi.org/10.1007/s42452-021-04427-5.

[40] M. Wang, J. Zhou, J. Gao, Z. Li, E. Li, Milling tool wear prediction method based on deep learning under variable working conditions, IEEE Access 8 (2020) 140726–140735, http://dx.doi.org/10.1109/ACCESS.2020.3010378.

[41] M. Shah, V. Vakharia, R. Chaudhari, J. Vora, D. Pimenov, K. Giasin, Tool wear prediction in face milling of stainless steel using singular generative adversarial network and LSTM deep learning models, Int. J. Adv. Manuf. Technol. 121 (2022) 723–736, http://dx.doi.org/10.1007/s00170-022-09356-0.

[42] J. Wang, J. Yan, C. Li, R. Gao, R. Zhao, Deep heterogeneous GRU model for predictive analytics in smart manufacturing: Application to tool wear prediction, Comput. Ind. 111 (2019) 1–14, http://dx.doi.org/10.1016/j.compind.2019.06.001.

[43] X. Liu, B. Zhang, X. Li, S. Liu, C. Yue, S. Liang, An approach for tool wear prediction using customized DenseNet and GRU integrated model based on multi-sensor feature fusion, J. Intell. Manuf. 34 (2023) 885–902, http://dx.doi.org/10.1007/s10845-022-01954-9.

[44] A. Vaswani, et al., Attention is all you need, in: Proc. 31st Int. Conf. Neural Inf. Process. Syst, 2017, pp. 6000–6010, http://dx.doi.org/10.5555/3295222.3295349.

[45] Z. Huang, J. Zhu, J. Lei, X. Li, F. Tian, Tool wear predicting based on multi-domain feature fusion by deep convolutional neural network in milling operations, J. Intell. Manuf. 31 (2020) 953–966, http://dx.doi.org/10.1007/s10845-019-01488-7.

[46] Z. Huang, J. Zhu, J. Lei, X. Li, F. Tian, Tool wear predicting based on multisensory raw signals fusion by reshaped time series convolutional neural network in manufacturing, IEEE Access 7 (2019) 178640–178651, http://dx.doi.org/10.1109/ACCESS.2019.2958330.

[47] X. Xu, J. Wang, B. Zhong, W. Ming, M. Chen, Deep learning-based tool wear prediction and its application for machining process using multi-scale feature fusion and channel attention mechanism, Meas. 177 (2021) 109254, http://dx.doi.org/10.1016/j.measurement.2021.109254.

[48] C. Yang, J. Zhou, E. Li, M. Wang, T. Jin, Local-feature and global-dependency based tool wear prediction using deep learning, Sci. Rep. 12 (2022) 14574, http://dx.doi.org/10.1038/s41598-022-18235-3.

[49] X. Xu, J. Wang, W. Ming, M. Chen, Q. An, In-process tap tool wear monitoring and prediction using a novel model based on deep learning, Int. J. Adv. Manuf. Technol. 112 (2021) 453–466, http://dx.doi.org/10.1007/s00170-020-06354-y.

[50] Z. Huang, J. Zhu, J. Lei, X. Li, F. Tian, Tool wear monitoring with vibration signals based on short-time Fourier transform and deep convolutional neural network in milling, Math. Probl. Eng. 2021 (2021) 1–14, http://dx.doi.org/10.1155/2021/9976939.

[51] C. Bishop, Christopher M., Mixture Density Networks, Tech. Rep., Univ. Aston, Birmingham, UK, 1994.

[52] H. Zen, A. Senior, Deep mixture density networks for acoustic modeling in statistical parametric speech synthesis, in: Proc. IEEE Int. Conf. Acoust. Speech & Signal Process, 2014, pp. 3844–3848, http://dx.doi.org/10.1109/ICASSP.2014.6854321.

[53] B. Lakshminarayanan, A. Pritzel, C. Blundell, Simple and scalable predictive uncertainty estimation using deep ensembles, in: Proc. 31st Int. Conf. Neural Inf. Process. Syst, 2017, pp. 6405–6416, http://dx.doi.org/10.5555/3295222.3295387.

[54] C. Blundell, J. Cornebise, K. Kavukcuoglu, D. Wierstra, Weight uncertainty in neural network, in: Proc. 32nd Int. Conf. Mach. Learn, 2015, pp. 1613–1622, http://dx.doi.org/10.5555/3045118.3045290.

[55] S. Farquhar, L. Smith, Y. Gal, Liberty or depth: Deep Bayesian neural nets do not need complex weight posterior approximations, in: Proc. 34th Int. Conf. Neural Inf. Process. Syst, 2020, pp. 4346–4357, http://dx.doi.org/10.5555/3495724.3496089.

[56] J. Karandikar, T. McLeay, S. Turner, T. Schmitz, Tool wear monitoring using naive Bayes classifiers, Int. J. Adv. Manuf. Technol. 77 (9) (2015) 1613–1626, http://dx.doi.org/10.1007/s00170-014-6560-6.

[57] S. Shurrab, A. Almshnanah, R. Duwairi, Tool wear prediction in computer numerical control milling operations via machine learning, in: Proc. 12th Int. Conf. Inf. Commun. Syst, 2021, pp. 220–227, http://dx.doi.org/10.1109/ICICS52457.2021.9464580.

[58] D. McParland, et al., Prediction of tool-wear in turning of medical grade cobalt chromium molybdenum alloy (ASTM F75) using non-parametric Bayesian models, J. Intell. Manuf. 30 (3) (2019) 1259–1270, http://dx.doi.org/10.1007/s10845-017-1317-3.

[59] H. Sun, J. Pan, J. Zhang, D. Cao, Non-linear Wiener process-based cutting tool remaining useful life prediction considering measurement variability, Int. J. Adv. Manuf. Technol. 107 (2020) 4493–4502, http://dx.doi.org/10.1007/s00170-020-05264-3.

[60] Y. Li, Y. Xiang, B. Pan, L. Shi, A hybrid remaining useful life prediction method for cutting tool considering the wear state, Int. J. Adv. Manuf. Technol. 121 (2022) 3583–3596, http://dx.doi.org/10.1007/s00170-022-09417-4.

[61] J. Wang, P. Wang, R. Gao, Enhanced particle filter for tool wear prediction, J. Manuf. Syst. 36 (2015) 35–45, http://dx.doi.org/10.1016/j.jmsy.2015.03.005.

[62] H. Hanachi, W. Yu, I. Kim, J. Liu, C. Mechefske, Hybrid data-driven physics-based model fusion framework for tool wear prediction, Int. J. Adv. Manuf. Technol. 101 (9) (2019) 2861–2872, http://dx.doi.org/10.1007/s00170-018-3157-5.

[63] J. Zhang, B. Starly, Y. Cai, P. Cohen, Y. Lee, Particle learning in online tool wear diagnosis and prognosis, J. Manuf. Process. 28 (2017) 457–463, http://dx.doi.org/10.1016/j.jmapro.2017.04.012.

[64] P. Wang, R. Gao, Adaptive resampling-based particle filtering for tool life prediction, J. Manuf. Syst. 37 (2015) 528–534, http://dx.doi.org/10.1016/j.jmsy.2015.04.006.

[65] L. Hao, L. Bian, N. Gebraeel, J. Shi, Residual life prediction of multistage manufacturing processes with interaction between tool wear and product quality degradation, IEEE Trans. Automat. Sci. Eng. 14 (2) (2016) 1211–1224, http://dx.doi.org/10.1109/TASE.2015.2513208.

[66] Y. Cheng, D. Wang, P. Zhou, T. Zhang, Model compression and acceleration for deep neural networks: The principles, progress, and challenges, IEEE Signal Process. Mag. 35 (2018) 126–136, http://dx.doi.org/10.1109/MSP.2017.2765695.

[67] B. Athiwaratkun, Jx. Stokes, Malware classification with LSTM and GRU language models and a character-level CNN, in: Proc. IEEE Int. Conf. Acoust. Speech & Signal Process, 2017, pp. 2482–2486, http://dx.doi.org/10.1109/ICASSP.2017.7952603.

[68] L. Liu, F. Wu, Y. Wang, J. Wang, Multi-receptive-field CNN for semantic segmentation of medical images, IEEE J. Biomed. Health Inform. 24 (11) (2020) 3215–3225, http://dx.doi.org/10.1109/JBHI.2020.3016306.

[69] S. Ioffe, C. Szegedy, Batch normalization: Accelerating deep network training by reducing internal covariate shift, 2015, [Online]. Available: http://arxiv.org/abs/1502.03167.

[70] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: a simple way to prevent neural networks from overfitting, J. Mach. Learn. Res. 15 (1) (2014) 1929–1958, http://dx.doi.org/10.5555/2627435.2670313.

[71] Y. Gal, Z. Ghahramani, Dropout as a Bayesian approximation: Representing model uncertainty in deep learning, in: Proc. 33rd Int. Conf. Mach. Learn, 2016, pp. 1050–1059, http://dx.doi.org/10.5555/3045390.3045502.

[72] A. Kendall, Y. Gal, What uncertainties do we need in Bayesian deep learning for computer vision? in: Proc. 31st Int. Conf. Neural Inf. Process. Syst, 2017, pp. 5580–5590, http://dx.doi.org/10.5555/3295222.3295309.

[73] D. Kingma, M. Welling, Auto-encoding variational Bayes, 2013, [Online]. Available: https://arxiv.org/abs/1312.6114.

[74] A. Graves, Practical variational inference for neural networks, in: Proc. 24th Int. Conf. Neural Inf. Process. Syst, 2011, pp. 2348–2356, http://dx.doi.org/10.5555/2986459.2986721.

[75] G. Hinton, D. Van Camp, Keeping the neural networks simple by minimizing the description length of the weights, in: Proc. 6th Annu. Conf. Comput. Learn. Theory, 1993, pp. 5–13, http://dx.doi.org/10.1145/168304.168306.

[76] M. Khan, A. Aravkin, M. Friedlander, M. Seeger, Fast dual variational inference for non-conjugate latent Gaussian models, in: Proc. Int. Conf. Mach. Learn, 2013, pp. 951–959, http://dx.doi.org/10.5555/3042817.3043043.

[77] M. Abdar, et al., A review of uncertainty quantification in deep learning: Techniques, applications and challenges, Inf. Fusion 76 (2021) 243–297, http://dx.doi.org/10.1016/j.inffus.2021.05.008.

[78] G. D'Mello, P. Pai, N. Puneet, Optimization studies in high speed turning of Ti-6Al-4V, Appl. Soft Comput. 51 (2017) 105–115, http://dx.doi.org/10.1016/j.asoc.2016.12.003.

[79] E. Usui, T. Shirakashi, T. Kitagawa, Analytical prediction of cutting tool wear, Wear 100 (1984) 129–151, http://dx.doi.org/10.1016/0043-1648(84)90010-3.

[80] T. Matsumura, T. Shirakashi, E. Usui, Identification of wear characteristics in tool wear model of cutting process, Int. J. Mater. Form. 1 (2008) 555–558, http://dx.doi.org/10.1007/s12289-008-0297-4.

[81] M. Binder, F. Klocke, D. Lung, Tool wear simulation of complex shaped coated cutting tools, Wear 330 (2015) 600–607, http://dx.doi.org/10.1016/j.wear.2015.01.015.

[82] K. Hosseinkhani, E. Ng, A hybrid experimental and simulation approach to evaluate the calibration of tool wear rate models in machining, Int. J. Adv. Manuf. Technol. 96 (2018) 2709–2724, http://dx.doi.org/10.1007/s00170-018-1687-5.

[83] K. Levenberg, A method for the solution of certain non-linear problems in least squares, Quart. Appl. Math. 2 (2) (1944) 164–168, http://dx.doi.org/10.1090/qam/10666.

[84] G. Kim, D. Shin, J. Choi, S. Lim, A deep learning-based cryptocurrency price prediction model that uses on-chain data, IEEE Access 10 (2022) 56232–56248, http://dx.doi.org/10.1109/ACCESS.2022.3177888.

[85] J. Bergstra, Y. Bengio, Random search for hyper-parameter optimization, J. Mach. Learn. Res. 13 (2012) 281–305, http://dx.doi.org/10.5555/2188385.2188395.
[86] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proc. IEEE Conf. Comput. Vision Pattern Recognit, 2016, pp. 770–778, http://dx.doi.org/10.1109/CVPR.2016.90.
[87] M. Abadi, et al., Tensorflow: Large-scale machine learning on heterogeneous distributed systems, 2016, [Online]. Available: https://arxiv.org/abs/1603.04467.
[88] D. Kingma, J. Ba, Adam: A method for stochastic optimization, 2014, [Online]. Available: https://arxiv.org/abs/1412.6980.
[89] H. Deng, G. Runger, E. Tuv, M. Vladimir, A time series forest for classification and feature extraction, Inform. Sci. 239 (2013) 142–153, http://dx.doi.org/10.1016/j.ins.2013.02.030.
[90] K. Cho, B. Merriënboer, D. Bahdanau, Y. Bengio, On the properties of neural machine translation: Encoder–decoder approaches, 2014, [Online]. Available: https://arxiv.org/abs/1409.1259.
[91] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Comput. 9 (8) (1997) 1735–1780, http://dx.doi.org/10.1162/neco.1997.9.8.1735.
[92] D. Snr, The correlation of vibration signal features to cutting tool wear in a metal turning operation, Int. J. Adv. Manuf. Technol. 19 (2002) 705–713, http://dx.doi.org/10.1007/s001700200080.
[93] H. Choi, D. Kim, J. Kim, J. Kim, P. Kang, Explainable anomaly detection framework for predictive maintenance in manufacturing systems, Appl. Soft Comput. 125 (2022) 109147, http://dx.doi.org/10.1016/j.asoc.2022.109147.
[94] B. Tama, M. Vania, S. Lee, S. Lim, Recent advances in the application of deep learning for fault diagnosis of rotating machinery using vibration signals, Artif. Intell. Rev. (2022) 1–43, http://dx.doi.org/10.1007/s10462-022-10293-3.