A GENERALIZED METHOD FOR FEATURIZATION OF MANUFACTURING SIGNALS, WITH APPLICATION TO TOOL CONDITION MONITORING

Max Ferguson
Stanford University
Stanford, CA, USA

Kincho H. Law
Stanford University
Stanford, CA, USA

Raunak Bhinge
Infinite Uptime, Inc.
Berkeley, CA, USA

Yung-Tsun Tina Lee
National Institute of Standards and Technology (NIST)
Gaithersburg, MD, USA

ABSTRACT

The application of machine learning techniques in the manufacturing sector provides opportunities for increased production efficiency and product quality. In this paper, we describe how audio and vibration data from a sensor unit can be combined with machine controller data to predict the condition of a milling tool. Emphasis is placed on the generalizability of the method to a range of prediction tasks in a manufacturing setting. Time series, audio, and acceleration signals are collected from a Computer Numeric Control (CNC) milling machine and discretized into blocks. Fourier transformation is employed to create generic power spectrum feature vectors. A Gaussian Process Regression model is then trained to predict the condition of the milling tool from the feature vectors. We highlight that this multi-step procedure could be useful for a range of manufacturing applications where the frequency content of a signal is related to a value of interest.

INTRODUCTION

The application of modern machine learning techniques to manufacturing processes provides an opportunity to increase productivity and improve overall product quality in traditional manufacturing lines [1]. The adoption of predictive models within the industrial value chain is part of a larger transition often referred to as the industrial internet, which promises to bring substantially increased operational effectiveness as well as the development of entirely new business models, services, and products [2].

In order to increase manufacturing productivity while reducing maintenance costs, it is crucial to develop more intelligent maintenance strategies, that can predict when maintenance should be performed [3]. Reliable tool-condition monitoring is likely to play an important role in the reactive maintenance strategies of future manufacturing facilities.

Numerous machine learning models have been proposed to optimize a range of tasks, from robotic control to machine failure detection [4,5]. One of the hurdles preventing adoption on a broader scale is that preprocessing raw data into relevant features is a subjective and difficult process [6]. In developing most machine learning models, a domain expert is given the task of carefully selecting a set of inputs, normally referred to as features, that yield optimum performance for the given prediction task. While such a technique has been considered the status quo for some time, recent progress in deep learning has demonstrated that automatic feature selection often yields superior performance than manual feature selection [7]. The recent popularization of Deep Recurrent Neural Networks provides a promising method of analyzing time series data [8]. However, the development of deep neural networks requires a large training dataset and tremendous computational power [9].

Researchers have previously demonstrated that the condition of a machine tool can be inferred from features of the vibration and audio signals [10,11]. A number of researchers have attempted to use the skew and kurtosis coefficients of the audio and acceleration time-series to predict the condition of the tool, but with mixed results [11–13]. Bukkapatnam et al. developed a tool wear prediction technique using an artificial neural network (ANN) with features inspired by the principles of nonlinear dynamics [14]. Sanjay et al. developed a model for predicting tool flank wear using ANNs [15]. The feed rates, spindle speeds, torques, machining times, and thrust forces were used to train the ANN model. Wu et al. reviewed these methods and demonstrated an alternative approach using random forests with a set of manually selected features [3]. A wide range of tool monitoring techniques have been reviewed by Dimla et al [16]. They concluded that existing techniques perform well on carefully selected experimental data, but there is a need for a multi-level system capable of handling unprocessed data.

While the application of machine learning to continuous time series data brings about its own difficulties, there are several characteristics of manufacturing that make it a perfect match for machine learning. First, manufacturing tends to be a repetitive process, and hence the time series signals from manufacturing often tend to be repetitive. Second, faults in the manufacturing process are likely to produce a different signal, and can be identified by comparing the time series signal against that from an operational product line.

In this paper, we outline a methodology for extracting information from a time series data source, with emphasis on generalizability. In this method, the time series signals are aggregated, transformed, and then classified. In the first step, we break the time series into a series of blocks, using instructions from the machine controller. In the tool wear example, each block corresponds to a single cutting action of the milling machine. However, the method only requires that the blocks represent a signal from some repeated part of the process. In the second step, we calculate the power spectral density (PSD) of each time series block using Fast Fourier Transform (FFT). Finally, we train a Gaussian Process Regression (GPR) model to predict tool condition based on the PSD vectors.

The remainder of the paper is organized as follows: In the first section, we review how time series audio and vibration data are collected from a Computer Numeric Control (CNC) machine, namely a Mori Seiki NVD1500DCG. In the next section, we describe how the time series data is divided into blocks using information from the machine controller. We then discuss how feature vectors are developed to represent the frequency content of the audio and acceleration sensors. We conclude by demonstrating that the frequency vectors contain information about the tool condition, and subsequently show that they can be used to predict tool condition using a GPR algorithm.

TIME SERIES DATA COLLECTION

In this section, we describe how time series acceleration and acoustic data were collected from a CNC milling machine, namely a Mori Seiki NVD1500DCG. As shown in Figure 1, a waterproof sensor unit from Infinite Uptime was attached to the vise of the milling machine.

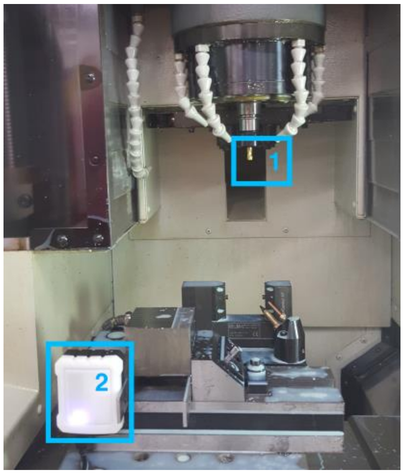


FIGURE 1. A MORI SEIKI NVD1500DCG MILLING MACHINE WITH CUTTING TOOL (1) AND SENSOR UNIT FROM INFINITE UPTIME (2)

The sensor unit was capable of measuring both the audio and triaxial acceleration signals inside the milling machine. The acceleration signal was recorded in the x-, y- and z-directions at 1000 Hz. The audio signal was recorded at 8000 Hz. Data is streamed from the sensor to a laptop computer using a Universal Serial Bus (USB) connection.

The milling machine was programmed to produce a number of simple 'parts' by removing material from a solid steel block. Each part consisted of 20 separate cutting actions performed by the milling machine. Figure 2 shows a section of acceleration time series data recorded using the sensor unit.

The machining data, such as tool position and rotation speed, were recorded from the FANUC controller. An MTConnect agent was used to synchronize the data from the milling machine and stream it to a laptop computer, along with a timestamp. A post-processing step was used to convert the machining data into a set of operations performed by the machine. The post-processing involved a simulation step, which is previously described in an earlier paper [17].

GENERIC FEATURE VECTORS

In machine learning, the process of featurization involves converting raw data into a vector form that is suitable for the chosen machine learning model. The featurization process often reduces the dimensionality of the data, which in turn reduces the computational burden of training the predictive model. In this section, we describe how the time series data is split into blocks using the controller data, and then converted into feature vectors.

In a more traditional application of machine learning, we would attempt to identify a range of different measures that correlate strongly with the target value of interest, which, in this study, is tool condition. A number of researchers have attempted to use the skew and kurtosis coefficients of the audio and acceleration time series to predict the condition of the tool with mixed results [11–13]. While this approach is valuable, it can also limit the application of the model to a very specific scenario. Instead, we create relatively large feature vectors, and use an optimization algorithm to assign a weight to each feature.

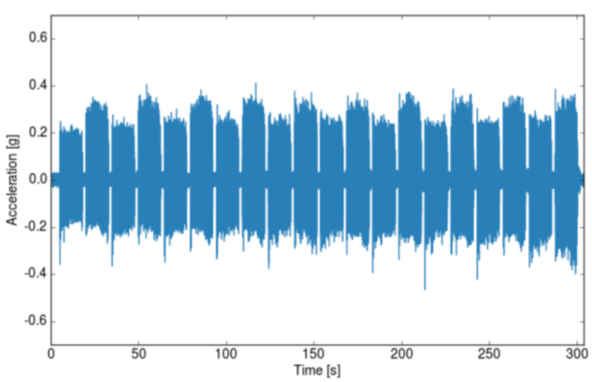


FIGURE 2. MEASURED ACCELERATION SIGNAL IN THE X-DIRECTION WHILE PRODUCING A SINGLE PART

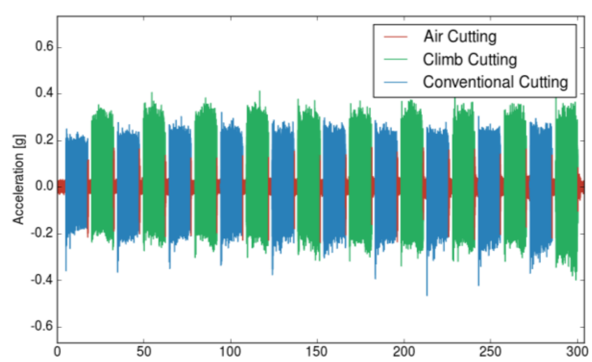


FIGURE 3. ACCELERATION SIGNAL IN THE X-DIRECTION, AFTER BEING AUTOMATICALLY LABELED USING THE CONTROLLER DATA.

Discretizing Time Series Data

The first step of preprocessing a time series data is to divide the data into blocks, according to the type of operation performed by the machine. The data from the milling machine controller is used to automatically label the time series data. Labelling the time series makes the dataset more structured, and thus makes the dataset more suitable for use with machine learning algorithms. The milling machine performs a number of different operations to produce a part. Figure 3 shows the time series data in the production of a part that involves 10 climb-cutting operations and 10 conventional-cutting operations. Each cutting operation is separated by a brief "air cutting" operation, in which the machine pauses briefly between cuts. We use the terms "climb cutting" and "conventional cutting" refer to the relationship between the rotation direction to the feed direction. A detailed description of these operations has been discussed earlier [17].

The audio and vibration signals produced are dependent on the type of cutting operation being performed by the milling machine, as illustrated in Figure 3.

Estimating the Power Spectral Density

Next, we estimate the power spectral density (PSD) of each time series block. As the audio and acceleration signals are periodic, the PSD provides an elegant way to summarize the information about the signals. In many cases, comparing the PSD of different blocks can reveal information about the underlying physical process.

FFT is used to estimate the PSD of each time series block. For each acceleration or audio signal, $s \in \mathbb{R}^N$, the PSD is given by:

$$\hat{s}(\omega) = \frac{1}{N}\left|\sum_{n=0}^{N-1} s(n)e^{-i\omega n}\right|^2, \tag{1}$$

where $\hat{s}(\omega)$ is the PSD of the signal at angular frequency ω. The PSD is computed at a discrete set of frequencies $\omega = 2\pi k/T$ for $k \in \{1 \dots N\}$, where T is the sampling period of the signal.

When the PSD is estimated in this manner the PSD vector $\hat{s} \in \mathbb{R}^N$, has the same length as the input time series signal $s \in \mathbb{R}^N$. Because inconsistencies exist in manufacturing processes, the length of each time series block varies slightly. This variation is typical of most manufacturing processes, where small variations tend to occur in each iteration of a repetitive process [18]. On the other hand, in machine learning it is expected that the length of the feature vectors is the same. Therefore, we calculate the PSD over sequential windows of constant length, and average the results. Specifically, the time series is broken into M consecutive segments, where each segment has a length of 256 points. The PSD is calculated for each of the M segments. The PSD for the entire block is obtained by averaging the PSDs for each segment. There are several benefits of computing the PSD in this manner. First, the length of the frequency coefficient vector is now the same across all time series blocks. Second, the averaging process helps to reduce noise in the PSD, while still providing a consistent estimate of the PSD [19].

To create a generic feature vector, the PSD vectors of each signal are combined. Let $\hat{\boldsymbol{a}}$ be the audio PSD and $\hat{\boldsymbol{v}}_x$, $\hat{\boldsymbol{v}}_y$ and $\hat{\boldsymbol{v}}_z$ be the vibration PSD. We denote the generic feature vector as:

$$\boldsymbol{x}^i = \begin{bmatrix} \hat{\boldsymbol{a}} \\ \hat{\boldsymbol{v}}_x + \hat{\boldsymbol{v}}_y + \hat{\boldsymbol{v}}_z \end{bmatrix}, \tag{2}$$

where the generic feature vector $\boldsymbol{x}^i$ contains the PSD from the audio and vibration signals. The PSD vector for the vibration data in each direction are added. By Parseval's theorem the result will always have the same signal energy as the sum of energies of the components [20]. In this way, the vibration component of the generic feature vector is largely invariant to the rotation of the accelerometer.

To demonstrate that the generic feature vector contains relevant information about the physical process, several feature vectors are compared. Figure 4 compares the feature vectors from operations with different cutting strategies. Figure 5 compares the coefficients from a new tool and a worn tool.

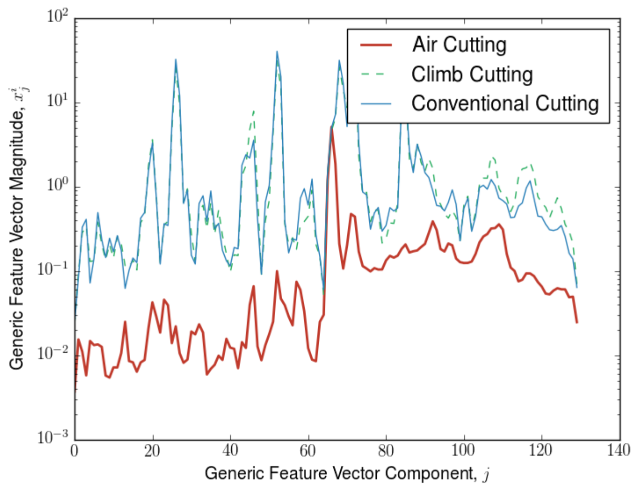


FIGURE 4. COMPARISON OF FEATURE VECTORS WHILE THE MACHINE WAS PERFORMING DIFFERENT CUTTING OPERATIONS.

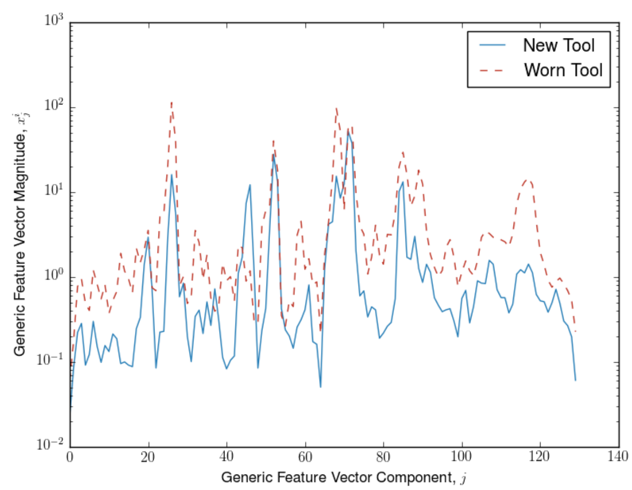


FIGURE 5. COMPARISON OF FEATURE VECTORS FOR A NEW TOOL AND A WORN TOOL, WHILE A CLIMB CUTTING OPERATION WAS BEING COMPLETED

TOOL CONDITION PREDICTION MODEL

In this section, we develop a GPR model to predict tool condition using the generic frequency feature vectors. A number of alternative techniques such as k-Nearest Neighbors and Support Vector Regression were considered. GPR is chosen as it performs particularly well with noisy data. An additional benefit of GPR is that it provides a distribution on the target value, as opposed to a scalar estimate.

GPR has been used to develop models for a range of manufacturing problems. In [21], GPR was used to predict the energy consumption of a CNC milling machine using features such as spindle speed and cutting type. A variant of GPR called Local Gaussian Process Regression (LGPR) was used in [4] to develop a model for real-time robot control. The Predictive Model Markup Language (PMML) was recently extended to offer GPR support, as described in [22] and [23], providing a standardized format to save and transport GPR models.

Data Collection

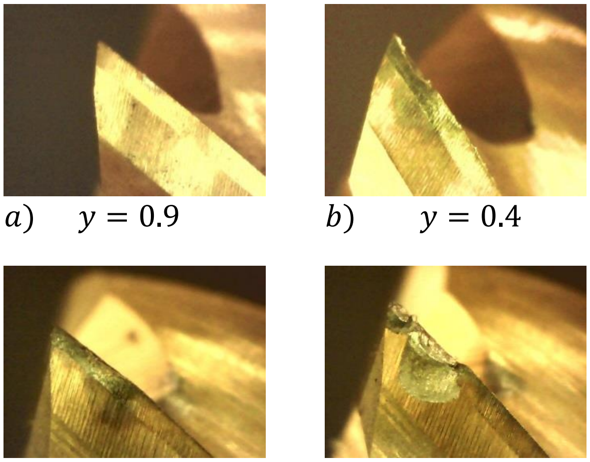
The Mori Seiki milling machine was used to produce parts until the cutting tool became severely damaged, or the cutting tool broke. A total of 14 tools were used to produce 56 parts. The audio and vibration time series were recorded as described earlier.

In a typical manufacturing environment, machine tools tend to last several days. To accelerate the testing process, the operating parameters of the machine were adjusted to increase the rate of tool-wear by increasing the feed rate and reducing the rotation speed. With the adjusted operating parameters, the operating lifetime of a cutting tool was reduced to about 30 minutes in this experiment.

Defining Tool Condition

A number of different methods have been proposed to measure the condition of a machine tool. While quantitative measurements such as the wear depth have been proposed as tool condition measures, these measures often fail to accurately capture wear when the blade becomes chipped.

In this study, we define the condition of the milling machine tool $y_c \in [0,1]$, based on the remaining lifetime of the tool, as estimated after manually examining the tool with a microscope. The scale is defined such that 100 % indicates a new tool in perfect condition, and 50 % indicates the condition at which the tool would be replaced in a commercial manufacturing operation. Figure 6 illustrates four different states of the machine tool flute.



$a)\quad y = 0.9$ $b)\quad y = 0.4$

$c)\quad y = 0.6$ $d)\quad y = 0.1$

FIGURE 6. TOOL FLUTE IN DIFFERENT STATES OF CONDITION. A LOWER VALUE OF y INDICATES MORE WEAR

Gaussian Process Model

In GPR, a Gaussian process (GP) is used as a prior to describe the distribution on the target function $y = f(x)$. A GP is a generalization of the Gaussian probability distribution for which any finite linear combination of samples has a joint Gaussian distribution [24].

A GP can be fully specified by its mean function m(·) and covariance kernel function k(·,·).

$$p(f^{1:n}) = GP\big(m(\cdot), k(\cdot,\cdot)\big). \qquad (3)$$

The mean function $m(\cdot)$ captures the overall trend in the target function value and the kernel function $k(\cdot,\cdot)$ is used to approximate the covariance by representing the similarity between the data points [24]. For the tool condition model, we choose to use the zero function as the mean function.

In general, we denote the input as $\boldsymbol{x}^i \in \mathbb{R}^{\boldsymbol{n}}$ and the target value as $y \in \mathbb{R}$. In the proposed data processing method, the input $\boldsymbol{x}^i$ is the generic feature vector. A GP is used as a prior to describe the distribution on the target function $y^i = f(\boldsymbol{x}^i)$. We attempt to learn the target function by incorporating prior knowledge captured in historical data. Suppose the current data set is denoted by $\boldsymbol{D} = \{(\boldsymbol{x}^i, y^i) | i = 1, \dots n\}$. With GPR, the measured output $y^{new} = f(\boldsymbol{x}^{new})$ corresponding to the new feature vector x^{new} and the historical outputs $y^{1:n}$ in the training data set follow a multivariate Gaussian distribution:

$$\begin{bmatrix} \boldsymbol{y}^{1:n} \\ f^{new} \end{bmatrix} \sim N\left(\boldsymbol{0}, \begin{bmatrix} \mathbf{K} & \boldsymbol{k} \\ \boldsymbol{k}^T & k(\boldsymbol{x}^{new}, \boldsymbol{x}^{new}) \end{bmatrix}\right), \qquad (4)$$

where the entries in the vector $\boldsymbol{k}$, and the covariance kernel matrix $\mathbf{K}$, are defined respectively as:

$$\boldsymbol{k}_i = k(\boldsymbol{x}^i, \boldsymbol{x}^{new}), \qquad (5)$$

$$\mathbf{K}_{ij} = k(\boldsymbol{x}^i, \boldsymbol{x}^j). \qquad (6)$$

The covariance kernel matrix $\mathbf{K}$ is often precomputed on the training data, allowing new predictions to be computed efficiently.

Selecting a Kernel Function

The covariance kernel function provides an efficient method to compute the similarity between two generic feature vectors. In GPR, the kernel function is used to estimate the covariance between two input vectors, $\boldsymbol{x}^{\mathrm{i}}$ and $\boldsymbol{x}^j$.

The Automatic Relevance Determination (ARD) squared exponential kernel is often used with GPR, as it automates the selection of feature weights. The ARD squared exponential function can be expressed as:

$$k(\boldsymbol{x}^i, \boldsymbol{x}^j) = \gamma \exp\left(-\frac{1}{2}(\boldsymbol{x}^i - \boldsymbol{x}^j)^T \mathrm{diag}(\boldsymbol{\lambda})^{-2}(\boldsymbol{x}^i - \boldsymbol{x}^j)\right), \qquad (7)$$

where the kernel function is described by the hyper-parameters, γ and $\boldsymbol{\lambda}$. The signal variance hyper-parameter γ quantifies the overall magnitude of the covariance value. The hyper-parameter λ_k where $k \in \{1 \dots |\boldsymbol{\lambda}|\}$ is used to quantify the relevancy of the input feature x_k^i when predicting the response y. During the training process, an optimization problem is constructed to maximize the likelihood of the training data, relative to the hyper-parameters γ and $\boldsymbol{\lambda}$ [24].

Noise Model

Each tool condition label y, is likely to contain random noise due to the complex nature of tool wear and the manual labelling method. To account for this random noise we assume that each observed value contains some random noise ϵ, such that $y = f(\boldsymbol{x}) + \epsilon$. We assume that this noise follows an independent, identically distributed Gaussian distribution with zero mean and variance σ_ϵ^2:

$$\epsilon \sim \mathcal{N}(0, \sigma_\epsilon^2). \quad (8)$$

It follows from the independence assumption that the noise model can be represented by adding a noise term to the kernel function [24]:

$$k(\boldsymbol{x}^i, \boldsymbol{x}^j) = \gamma \exp\left(-\frac{1}{2}(\boldsymbol{x}^i - \boldsymbol{x}^j)^T \operatorname{diag}(\boldsymbol{\lambda})^{-2}(\boldsymbol{x}^i - \boldsymbol{x}^j)\right) + \sigma_\epsilon^2 \delta_{ij}, \quad (9)$$

where δ_{ij} represents Kronecker delta function which serves to selectively add the noise variance σ_ϵ^2 to the covariance value.

Model Regularization

The size of the hyperparameter vector is dependent on the choice of kernel and the length of the input vector $\boldsymbol{x}^i$. With the ARD squared exponential kernel, the number of hyperparameters can become reasonably large:

$$|\boldsymbol{\theta}| = |\gamma| + |\boldsymbol{\lambda}| = 1 + |\boldsymbol{x}^i|. \quad (10)$$

The large number of hyperparameters increases the flexibility of the model, allowing it to represent high-dimensional relationships. The increased model complexity also makes it prone to overfitting. In machine learning, regularization is commonly used to limit the parameter space of the model. Bayesian model selection is an alternative approach to regularization, used to constrain the model complexity, as described in [25]. We start by defining a prior $p(\boldsymbol{\theta})$ on the hyperparameters $\boldsymbol{\theta}$, to restrict the hyperparameter space. Hereafter we will refer to $p(\boldsymbol{\theta})$ as the hyperprior, as is common in existing literature [24]. In the proposed methodology, we choose the hyperprior to minimize the weights in $\boldsymbol{\theta}$.

$$p(\boldsymbol{\theta}) \sim \mathcal{N}(0, \alpha^2 \boldsymbol{I}), \quad (11)$$

where α is a regularization parameter which controls the flexibility of the model. For a large α, the model will tend to fit to the training data well, but is unlikely to generalize well to the unseen test data. As the value of α is decreased the model will become more constrained, and the generalization error will tend to decrease. The optimum value of α is found through cross-validation.

Training Procedure

In the training procedure a GP is fitted to the training data set. Suppose we denote the dataset $\boldsymbol{D} = \{(\boldsymbol{x}^i, y^i) | i = 1, \ldots n\}$ for time series block i. The GPR training procedure involves selecting a set of hyperparameters to maximize the marginal likelihood of the training data. The marginal distribution of the observations can be expressed as:

$$p(\boldsymbol{y}^{1:n}|\boldsymbol{\theta}) = \int p(\boldsymbol{y}^{1:n}|\boldsymbol{f}^{1:n})\, p(\boldsymbol{\theta}) p(\boldsymbol{f}^{1:n}|\boldsymbol{\theta})\, d\boldsymbol{f}^{1:n}. \quad (12)$$

The unknown function $\boldsymbol{f}$ can be marginalized out of (12) to obtain the marginal likelihood of the training observations. The hyperparameters $\boldsymbol{\theta}$ are chosen to maximize the marginal likelihood of observations in a given training data set $\boldsymbol{D}$. An optimization equation is formed to maximize the marginal likelihood, and obtain to the optimum hyperparameters $\boldsymbol{\theta}^*$:

$$\boldsymbol{\theta}^* = \arg\max_{\theta} \log p\,(\mathbf{y}^{1:\mathrm{n}}|\boldsymbol{\theta}\,). \quad (13)$$

Finding the optimum hyperparameters using (13) requires an iterative approach as the value of the kernel matrix $\mathbf{K}$ is inherently dependent on the hyperparameters. The process for obtaining the optimum hyperparameters is well documented in the literature [24]. The MATLAB library GPML [25] is chosen to optimize the hyperparameters.

Scoring Procedure

In GPR, the aim of the scoring procedure is to obtain a posterior distribution f^{new} on the output value, based on the previously unseen observation $\boldsymbol{x}^{new}$. In the case where the mean function is zero, the posterior distribution on the response f^{new} for the newly observed input $\boldsymbol{x}^{new}$ can be expressed as a Gaussian distribution:

$$f^{new} \sim \mathcal{N}\left(\mu(\boldsymbol{x}^{new}|\boldsymbol{D}^n), \sigma^2(\boldsymbol{x}^{new}|\boldsymbol{D}^n)\right). \quad (14)$$

The posterior mean $\mu(\boldsymbol{x}^{new}|\boldsymbol{D}^n)$, and associated variance $\sigma^2(\boldsymbol{x}^{new}|\boldsymbol{D}^n)$, can be calculated directly. As the posterior distribution is 1D Gaussian, the posterior mean and variance are sufficient to fully describe the posterior distribution. That is, the posterior distribution can be expressed as [24]:

$$\mu(\boldsymbol{x}^{new}|\boldsymbol{D}^n) = \boldsymbol{k}^T(\mathbf{K} + \sigma_\epsilon^2 \mathbf{I})^{-1}\boldsymbol{y}^{1:n}, \quad (15)$$

$$\sigma^2(\boldsymbol{x}^{new}|\boldsymbol{D}^n) = k(\boldsymbol{x}^{new}, \boldsymbol{x}^{new}) - \boldsymbol{k}^T(\mathbf{K} + \sigma_\epsilon^2 \mathbf{I})^{-1}\boldsymbol{k}. \quad (16)$$

MODEL PERFORMANCE

When evaluating machine learning models, it is common practice to divide the data set into a training set and testing set. The model is trained on the training set and then tested on the previous unseen data in the testing set. Data points from three cutting tools were randomly selected for the testing set. Two GP models are trained to predict tool condition; the first is trained with the generic feature vectors from the climb-cutting blocks, and the second is trained with the feature vectors from the conventional-cutting blocks.

The two models are used together to predict the condition of the tool, for each point in the testing set. Figure 7 and Figure 8 provide a comparison of the model predictions with the human labels. In the ideal case, the model would predict the same values as the human labels, and the plotted points would align with the dotted diagonal line. It can be observed that the trained model predicts the tool conditions comparable to the human labeled results.

The tool condition predicted by the model closely aligns with the human labelled tool condition for relatively new tools, especially for tool condition in the 90-100 % range. This is most likely because the new tools provide a very consistent audio and vibration signal. Once the tool condition drops below 40 % there is a larger variation in the audio and vibration signals, making tool condition prediction more difficult. For example, a cutting tool at 40 % condition may have sustained heat damage and worn smooth, or it may have undergone brittle failure and chipped. Both failure modes would provide different audio and vibration signals, but the model is expected to label both cases as 40 % wear. The confidence interval increases as the condition decreases, indicating that it is harder to predict more heavily worn tools, as shown in Figure 7 and Figure 8.

It is likely that the accuracy of the model could be improved by increasing the amount of data in the training data set. The training data set does not contain many time series segments for worn tools, as a number of tests had to be stopped before the tool condition dropped below 20 %. Increasing the number of training data points collected with worn tools could reduce the confidence intervals in the predictions.

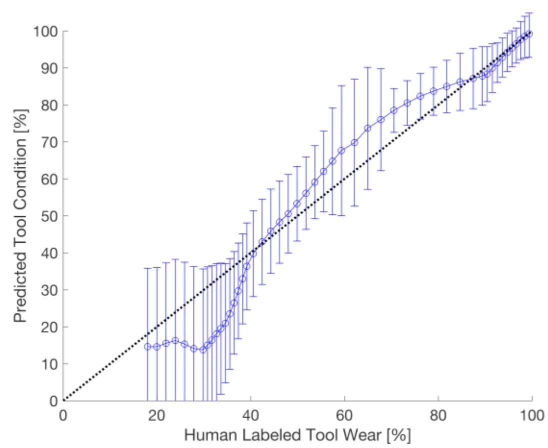


FIGURE 7. TOOL CONDITION PREDICTION FOR THE CLIMB CUTTING ACTIONS IN ONE OF THE THREE TEST DATASETS. THE ERROR BARS INDICATE ONE STANDARD DEVIATION IN THE TARGET DISTRIBUTION.

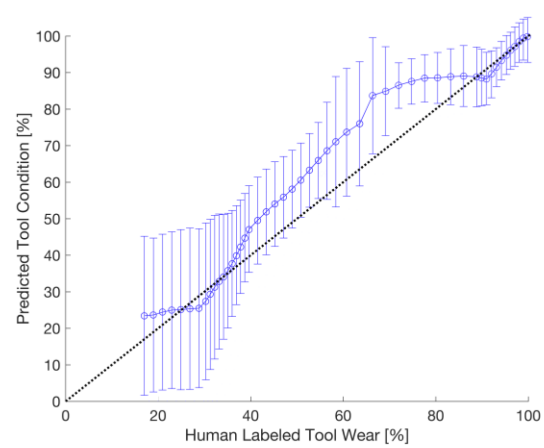


FIGURE 8. TOOL CONDITION PREDICTION FOR THE CONVENTIONAL CUTTING ACTIONS IN ONE OF THE THREE TEST DATASETS. THE ERROR BARS INDICATE ONE STANDARD DEVIATION IN THE TARGET DISTRIBUTION.

However, in an industrial application the cutting tool will be replaced before it reaches 50 % wear, so the accuracy of the model is less critical for heavily worn tools.

DISCUSSION

This study demonstrates how information from a milling machine controller can be combined with sensor data to predict the condition of the milling machine tool. Information from the milling machine controller was used to aggregate the time series data over a set of finite intervals. The frequency content in each time series block were summarized using a PSD. The PSD from the audio and vibration signals were combined to create a generic feature vector, containing information about each time series block.

The use of a non-parametric regression model, namely GPR, allowed the complex relationship between the generic feature vector and the target value to be modelled. The ARD squared exponential kernel was used to automate the feature selection process. The combination of the generic feature vectors and automated weighting procedure make this technique applicable to a range of different modelling tasks in the manufacturing domain. However, the cross-validation training procedure increases the training time by an order of magnitude.

The GP model provides confidence bounds for the predictive estimations, which are useful when interpreting the reliability of a prediction at some arbitrary time. The confidence bounds would likely prove useful in a practical application where the tool-condition predictions were used to determine when to change machine tools.

FUTURE WORK

For this method to have practical applications, it must generalize well to a range of different machines and machine operations. A similar technique could be applied to predict tool condition on a range of different manufacturing machines. The same technique could also be applied to correlate time series signals with surface quality or bearing failure.

ACKNOWLEDGMENT

The authors acknowledge the support by the Smart Manufacturing Systems Design and Analysis Program at the National Institute of Standards and Technology (NIST), US Department of Commerce. This work was performed under the financial assistance award (NIST Cooperative Agreement 70NANB17H031) to Stanford University. The authors would like to acknowledge the collaboration with the late Professor David Dornfeld of University of California Berkeley, who was instrumental in making the experiments possible.

DISCLAIMER

Certain commercial systems are identified in this paper. Such identification does not imply recommendation or endorsement by NIST; nor does it imply that the products identified are necessarily the best available for the purpose. Further, any opinions, findings, conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of NIST or any other supporting U.S. government or corporate organizations.

REFERENCES

[1] Monostori, L., 2003, “AI and Machine Learning Techniques for Managing Complexity, Changes and Uncertainties in Manufacturing,” Engineering Applications of Artificial Intelligence, **16**(4), pp. 277–291.

[2] Hermann, M., Pentek, T., and Otto, B., 2016, “Design Principles for Industrie 4.0 Scenarios,” *49th Hawaii International Conference on Hawaii International Conference on System Sciences (HICSS)*, IEEE, pp. 3928–3937.

[3] Wu, D., Jennings, C., Terpenny, J., Gao, R. X., and Kumara, S., 2017, “A Comparative Study on Machine Learning Algorithms for Smart Manufacturing: Tool Wear Prediction Using Random Forests,” Journal of Manufacturing Science and Engineering, **139**(7), p. 71018.

[4] Nguyen-Tuong, D., and Peters, J., 2008, “Local Gaussian Process Regression for Real-Time Model-Based Robot Control,” *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, pp. 380–385.

[5] Tai, A. H., Ching, W.-K., and Chan, L.-Y., 2009, “Detection of Machine Failure: Hidden Markov Model Approach,” Computers & Industrial Engineering, **57**(2), pp. 608–619.

[6] Du, K.-L., and Swamy, M., 2013, *Neural Networks and Statistical Learning*, Springer Science & Business Media.

[7] Wollsen, M. G., Hallam, J., and Jørgensen, B. N., 2016, “Novel Automatic Filter-Class Feature Selection for Machine Learning Regression,” *INNS Conference on Big Data*, Springer, pp. 71–80.

[8] Zhao, R., Yan, R., Chen, Z., Mao, K., Wang, P., and Gao, R. X., 2016, “Deep Learning and Its Applications to Machine Health Monitoring: A Survey,” submitted to IEEE Transactions on Neural Networks and Learning Systems.

[9] Aggarwal, C. C., 2014, *Data Classification: Algorithms and Applications*, Chapman & Hall/CRC.

[10] Kannatey-Asibu, E., and Dornfeld, D. A., 1982, “A Study of Tool Wear Using Statistical Analysis of Metal-Cutting Acoustic Emission,” Wear, **76**(2), pp. 247–261.

[11] Silva, R., Reuben, R., Baker, K., and Wilcox, S., 1998, “Tool Wear Monitoring of Turning Operations by Neural Network and Expert System Classification of a Feature Set Generated from Multiple Sensors,” Mechanical Systems and Signal Processing, **12**(2), pp. 319–332.

[12] Diniz, A., Liu, J., and Dornfeld, D., 1992, “Correlating Tool Life, Tool Wear and Surface Roughness by Monitoring Acoustic Emission in Finish Turning,” Wear, **152**(2), pp. 395–407.

[13] Dimla, D. E., 2000, “Sensor Signals for Tool-Wear Monitoring in Metal Cutting Operations—a Review of Methods,” International Journal of Machine Tools and Manufacture, **40**(8), pp. 1073–1098.

[14] Moon, F. C., and Kalmár-Nagy, T., 2001, “Nonlinear Models for Complex Dynamics in Cutting Materials,” Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences, **359**(1781), pp. 695–711.

[15] Sanjay, C., Neema, M., and Chin, C., 2005, “Modeling of Tool Wear in Drilling by Statistical Analysis and Artificial Neural Network,” Journal of Materials Processing Technology, **170**(3), pp. 494–500.

[16] Dimla, D., Lister, P., and Leighton, N., 1997, “Neural Network Solutions to the Tool Condition Monitoring Problem in Metal Cutting—a Critical Review of Methods,” International Journal of Machine Tools and Manufacture, **37**(9), pp. 1219–1241.

[17] Bhinge, R., Biswas, N., Dornfeld, D., Park, J., Law, K. H., Helu, M., and Rachuri, S., 2014, “An Intelligent Machine Monitoring System for Energy Prediction Using a Gaussian Process Regression,” *2014 IEEE International Conference on Big Data*, pp. 978–986.

[18] Braha, D., 2013, *Data Mining for Design and Manufacturing: Methods and Applications*, Springer Science & Business Media.

[19] Hayes, M. H., 2009, *Statistical Digital Signal Processing and Modeling*, Wiley India Limited.

[20] Landin, N., Romano, J. M., McMahan, W., and Kuchenbecker, K. J., 2010, “Dimensional Reduction of High-Frequency Accelerations for Haptic Rendering,” *International Conference on Human Haptic Sensing and Touch Enabled Computer Applications*, Springer, pp. 79–86.

[21] Park, J., Law, K. H., Bhinge, R., Biswas, N., Srinivasan, A., Dornfeld, D. A., Helu, M., and Rachuri, S., 2015, “A Generalized Data-Driven Energy Prediction Model with Uncertainty for a Milling Machine Tool Using Gaussian Process,” *ASME 2015 International Manufacturing Science and Engineering Conference*, American Society of Mechanical Engineers.

[22] Ferguson, M., Law, K. H., Park, J., Bhinge, R., Dornfeld, D. A., and Lee, Y.-T. T., 2016, “Evaluation of a PMML-Based GPR Scoring Engine on a Cloud Platform and Microcomputer Board for Smart Manufacturing,” *2016 IEEE International Conference on Big Data*.

[23] Park, J., Lechevalier, D., Ak, R., Ferguson, M., Law, K. H., Lee, Y.-T. T., and Rachuri, S., 2017, “Gaussian Process Regression (GPR) Representation Using Predictive Model Markup Language (PMML),” Smart and Sustainable Manufacturing Systems. Manuscript submitted for publication.

[24] Rasmussen, C. E., and Williams, C. K., 2006, *Gaussian Processes for Machine Learning*.

[25] “Documentation for GPML Matlab Code” [Online].Available: http://www.gaussianprocess.org/ gpml/code/matlab/doc/