Proceedings of the ASME 2023
Manufacturing Science and Engineering Conference
MSEC2023
June 12–16, 2023, New Brunswick, NJ

MSEC2023-105170

EDGE COGNITIVE DATA FUSION: FROM IN-SITU SENSING TO QUALITY CHARACTERIZATION IN HYBRID MANUFACTURING PROCESS

Danny Hoang[1], Nasir Mannan[2], Ruby ElKharboutly[3], Ruimin Chen[1], Farhad Imani[1]

[1]Department of Mechanical Engineering, University of Connecticut, Storrs, CT
[2]Connecticut Center for Advanced Technology, East Hartford, CT
[3]School of Computing and Engineering, Quinnipiac University, Hamden, CT

ABSTRACT

While hybrid additive manufacturing offers improved material properties, increased design flexibility, and reduced production time, the presence of variation in the process (e.g., torque, current, power, and tool speed), together with the impact of tool degradation (e.g., spindles, holders, and cutters) alters the surface roughness and dimensional accuracy of fabricated parts. Recently, edge sensors are coupled with machine learning techniques (e.g., feature-based support vector machines and end-to-end deep neural networks) to connect process parameters with build quality. However, the lack of interpretable learning, poor sample efficiency, and low accuracy and precision limit their capabilities for reliable analysis. This paper introduces hyperdimensional computing (HDC) to fuse load, current, torque, command speed, control differential, power, and contour deviation which provides robust, sample-efficient, and explainable learning of quality characterization. Experimental results on a real-world hybrid 5-axis CNC Deckel-Maho-Gildemeister (DMG) machine show that HDC achieves a superior 90.5% accuracy in prediction of deviation for a 25.4 mm counterbore feature using multichannel data. Comparing the difference in results with ML methods such as support vector machines, logistic regression, multinomial Naive-Bayes, multilayer perceptron, and residual neural network, HDC outperforms by 55.8%, 28.6%, 53.1%, 11.6%, and 28.0%, respectively. The proposed HDC is shown to be effective for data fusion and training with relatively few iterations and eliminates the necessity of costly and long retraining in various manufacturing processes such as 3D printing and bio-fabrication.

Keywords: Hybrid additive manufacturing, edge computing, cognitive learning, data fusion

1. INTRODUCTION

Additive manufacturing has been widely utilized to manufacture parts directly from the digital design in a layer-by-layer fashion [1–3]. Particularly, additive techniques are capable of creating parts that are fully customized with complex internal structures. The direct energy deposition (DED) is one additive manufacturing process that involves melting and fusing material (e.g., powder or wire) as it is being deposited into the area of focus [4, 5]. Compared to the complex laser powder bed fusion (LPBF) where parts are created layer-by-layer, DED implements line-by-line fabrication which allows the use of heterogeneous powders and wires within the same part. This allows the creation of heterogeneous materials with specific desirable properties. In recent years, these advantages of DED have caused an increased focus on research and implementation in manufacturing with specific applications with hybrid additive manufacturing (HAM).

HAM is an emerging novel technique that combines additive manufacturing with a conventional manufacturing process and provides numerous benefits compared to a single manufacturing process [6]. Particularly, instead of traditional subtractive techniques where a whole block is machined to create the necessary part, additive manufacturing is used to create the part layer-by-layer, or line-by-line, with subtractive techniques employed afterwards [7, 8]. The process enables fabrication and part correction sequentially with little downtime as both processes are conducted within the same machine, reducing waste and lowering production costs [9].

However, the process of creating parts using a complex hybrid process may result in various flaws, such as errors in machining and defects on the surface. Since HAM involves employing computer numerical control (CNC) to correct parts, the process includes generating G-code for the part surface, determining process parameters such as spindle rate and nominal feed rate, calculating feed rate and positioning control, and executing the machine tool [10]. Multiple factors such as changes in feed rate, spindle speed, or machine tool conditions cause significant variation and degrade the quality of the final product. The defects in HAM are attributed to a range of aspects, including thermal distortion of machine tools, geometrical deviation of moving axes,

and tool deflections.

With rapid advances in sensing technology, HAM is shifting from post-build quality inspection to in-process measurements of various process factors such as power, torque, load, and command speed. As such, multichannel measurements become readily available and bring a data-rich environment in HAM [11]. This provides an unprecedented opportunity to improve HAM process control. Furthermore, edge architectures are recently integrated for real-time computation in sensor nodes while connecting data to the cloud. Machine learning (ML) methods such as feature-based support vector machines and end-to-end deep neural networks are recently coupled with in-situ sensing to link process parameters to part quality for the correction of process drift and part anomalies toward first-time-right fabrication. Research was performed to analyze energy consumption and health monitoring of tools through ML methods coupled with in-process data [12–14].

However, the current solutions still face the following challenges: (1) poor sample efficiency: Multifarious production conditions of HAM result in limited data for individual parts. As a result, the limited training data available to perform learning causes overfitting problems and prevents the development of a generalizable model. (2) lack of interpretable learning: The popular deep neural networks (e.g., convolutional neural networks, recurrent neural networks, and transformers) are black-box, which suffers from reliability and trust issues. (3) low accuracy and precision with high computational costs: The accuracy of ML methods such as Bayesian models with multichannel data depends on their parameterizations and require a large number of iterations.

This research introduces hyperdimensional computing (HDC) that mimics human brain function to fuse load, current, torque, command speed, control differential, power, and contour deviation data to provide robust, sample-efficient, and interpretable learning of quality characterization. HDC is motivated by the awareness of how the human brain operates on high-dimensional data representations [15–17]. In-process data is accordingly encoded with high-dimensional vectors, called hypervectors, which have thousands of components [18–21]. HDC incorporates learning capacity and mimics several crucial functions of the human memory model with vector operations, which are computationally manageable and mathematically stringent in describing human cognition: (1) being extremely parallel and suitable for in-situ learning with low resource utilization on lightweight devices, (2) revealing hidden features; enabling single-pass learning with limited data, (3) being robust to noise and corrupt data which is an essential feature for HAM systems, and (4) providing an intuitive and human-interpretable model [22, 23]. We conduct experiments on a hybrid 5-axis CNC Deckel-Maho-Gildemeister (DMG) machine and perform analysis through the proposed HDC to predict deviation in a 25.4 mm counterbore feature using multichannel data.

The remainder of this paper is organized as follows: Section 2 provides comprehensive literature reviews of the related studies on hybrid additive manufacturing process and limitations of the current methodologies. Section 3 introduces the HDC framework for model implementation and parameter tuned training. Section 4 discusses the detailed setup for the experiments. Section 5 illustrates experimental results. Finally, the conclusion of this proposed study is presented in Section 6.

2. RESEARCH BACKGROUND

Despite the advantages of hybrid additive manufacturing in terms of time, cost, and efficiency, in-process parameters such as feed rate, cutting speed, spindle speed, and depth of cut still impact the quality of the build during the CNC milling process. The occurrence of common errors, i.e., tool wear, tool breakage, tool deflection, and chatter, can lead to part deviation, poor surface finishes, or in extreme cases scrapped parts. Therefore, there is an urgent need to determine the optimum or suitable parameters to minimize such machining errors.

Research investigations have shown the current developments of in-process monitoring with various algorithms for investigating machining errors or part quality. Tangjitsitcharoen conducted in-process monitoring utilizing the power spectrum density of dynamic cutting force measured during cutting for the detection of chip formation and chatter [24]. Results showed that their algorithm was sufficient in detecting the status of the cutting process. Lamraoui et al. were able to implement support vector machines on only motor current signals to detect chatter in CNC milling [25]. However, achieving higher and more robust prediction in chatter classification is possible through multi-signal fusion. Zhang et al. incorporated a hybrid model incorporating both the Hidden Markov Model (HMM) and an artificial neural network (ANN) for chatter monitoring [26]. Experimental results on a lathe showed that their model was good at forecasting chatter. Sun and Zhang also incorporated HMM in forecasting chatter and it worked well, but the robustness of their model is unknown [27].

Patange and Jegadeeshwara extracted attributes from vibration data during face milling using the J48 decision tree algorithm and then classified the condition of tool inserts using viz. Bayes Net and Naive-Bayes [14]. Pantange and Jegadeeshwara achieved relatively high accuracies for both Bayesian algorithms. Chungchoo and Saini used a fuzzy neural network with parameters such as cutting force, feed force, speed, and feed rate to predict flank and crater wear and were able to get accurate results [28]. However, Chungchoo and Saini acknowledge that their model involves training with data collected under specific lab conditions rather than more realistic conditions, which can affect their accuracy on tool wear estimation. Furthermore, [29–31] all implemented deep learning ANN for tool wear monitoring and all showed great prediction in tool wear, especially with the model in [30] outperforming traditional ML methods with manual feature extraction.

Zhang and Shetty incorporated a least squares support vector machine in predicting surface roughness in a machined surface and were able to outperform the analysis of variance method and a neural network [32]. Additionally, [33–35] also incorporated support vector machines for surface roughness prediction and all achieved low errors in their prediction. Wu and Lei used an ANN with features extracted from vibration signals and their corresponding envelopes to predict surface roughness of S45C steel in milling [36]. They showed that incorporating both cutting parameters and in-process vibrations signals enhances the accuracy of surface roughness prediction using ANN. Lin et al. evaluated

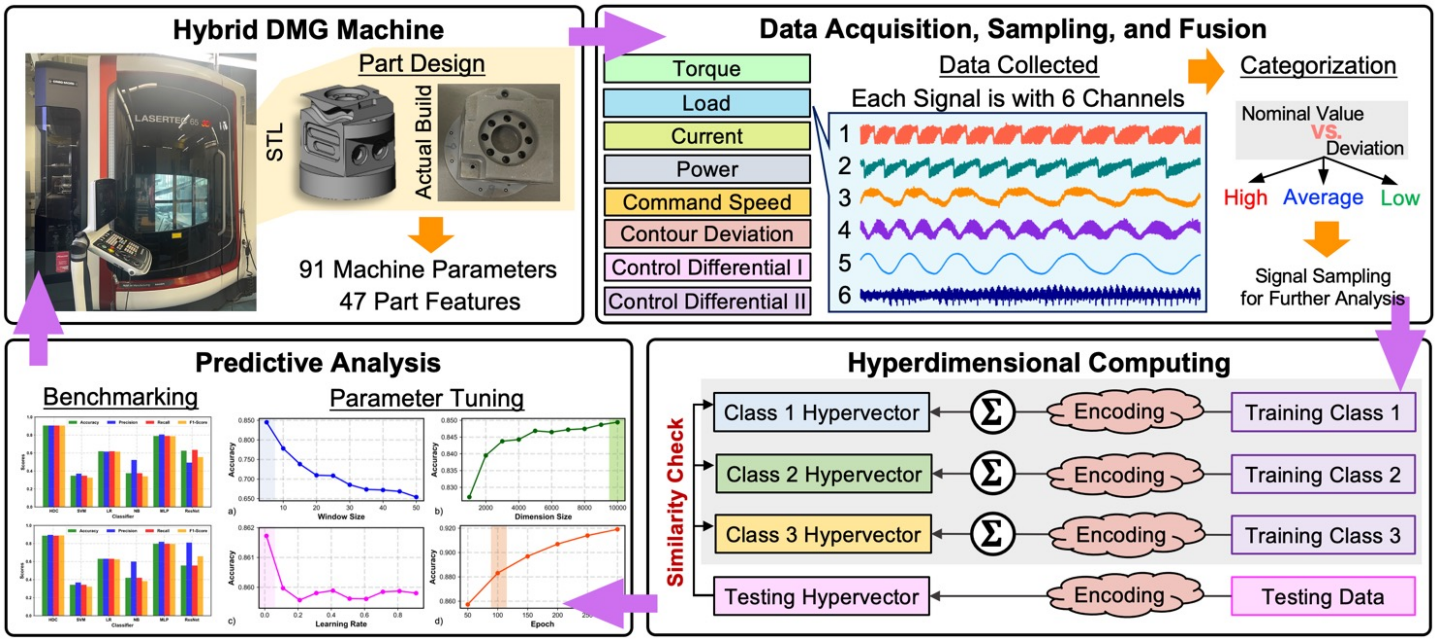

FIGURE 1: FLOWCHART OF THE PROPOSED FRAMEWORK FOR DATA FUSION AND COGNITIVE HDC FRAMEWORK FOR CHARACTERIZATION OF PART QUALITY IN HYBRID ADDITIVE MANUFACTURING PROCESS.

deep learning neural networks with vibrational data for surface roughness prediction [37]. They reported that a one-dimensional convolutional neural network can extract features automatically and can extract more information compared to manual extraction using a Fast Fourier Transform.

The use of few process parameters, or process signatures, in these many analyses lead more to be desired. By incorporating few, or in some instances just one parameter or signature, underlying phenomena can not be characterized to their full extent. The omission of signals removes important manufacturing phenomena that can better describe the various manufacturing errors experienced during fabrication. Furthermore, many of these models are trained with few samples, which can lead to overfitting and lack of applications beyond the experiment in itself. Many experiments were also able to achieve relatively high accuracies in their prediction but fail to test or report the robustness of their model. Additionally, many papers have demonstrated that automatic feature extraction in their models allows greater performance compared to models with manual feature extraction. Therefore, new algorithms are needed to address the barriers in the robust learning.

3. RESEARCH METHODOLOGY

This research characterizes the quality of parts in the hybrid manufacturing process from multiple in-situ data. As illustrated in Figure 1, 18 parts are manufactured by the CNC system of DMG hybrid manufacturing machine. Multiple process signals e.g., load, current, torque, command speed, control differential, power, and contour deviation, are recorded during the fabrication of the part using a Sinumerik edge device. After fabrication, 47 features are measured from the part for further analysis. The deviation of a 25.4 mm counterbore is selected as the representation of part quality as it has the most amount of deviation from the nominal value. A novel HDC framework is then proposed to combine 48 process channels to determine the relationship between part quality and multichannel process signature. In the end, HDC is compared with other state-of-the-art ML methods based on performance metrics such as accuracy, precision, recall, and F1-score to verify its superior learning capability.

The HDC framework is inspired by how the human brain abstractly represents objects and information. It involves two main stages: encoding and learning, as illustrated in Figure 2. During the encoding stage, the training data is transformed into a high-dimensional space. For data fusion, multiple pieces of data are combined and then encoded into a hypervector with a length of thousands. Hypervectors related to each training class are generated using bundling (element-wise addition). Finally, the similarity between the class hypervectors and encoded query data is compared to classify the data. The model can undergo iterative training to improve its performance.

3.1 Encoding

The first step in the described process is the creation of a hyperdimensional space using quasi-orthogonal basis hypervectors, denoted as $\mathbf{G} = \{\mathbf{G}_1, \mathbf{G}_2, ..., \mathbf{G}_m\}$, where $\mathbf{G}_i \, \forall i \in 1, ..., m$ has dimensionality D. In basis vector generation, each element of the hypervector is drawn from a standard normal distribution. The resulting hypervector is a high-dimensional vector that can be used to encode data to the dimensions of the vector. Since each element of $\mathbf{G}$ is randomly sampled from a standard normal distribution, and the dimension of the vector is large, two basis hypervectors are nearly orthogonal. In other words, $\delta(\mathbf{G}_i, \mathbf{G}_j) \simeq 0 \quad (0 < i, j < m, i \neq j)$.

Next, we define a set of instances $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N\}$ with corresponding labels $\mathbf{Y} = \{y_1, y_2, ..., y_N\}$. The instances are created by defining a window over signals, and the goal is to find

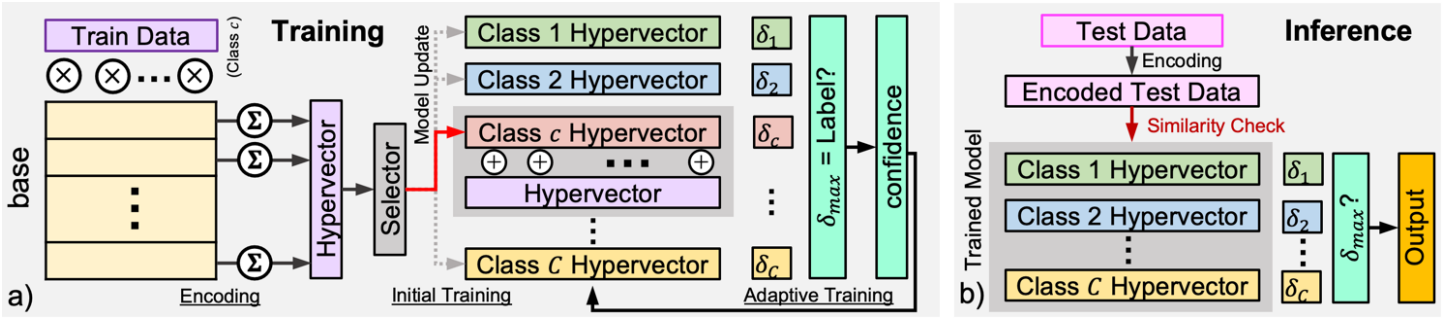

FIGURE 2: OVERVIEW OF THE PROPOSED HDC FRAMEWORK. A) DATA ENCODING, MODEL INITIAL TRAINING, AND MODEL ADAPTIVE TRAINING. B) MODEL INFERENCE.

the relationship between data input and its corresponding labels in the hyperdimension. To achieve this, we map each instance in $\mathbf{X}$ to a high-dimensional space and form a set of hypervectors $\mathbf{Q} = \{\mathbf{Q}_1, \mathbf{Q}_2, ..., \mathbf{Q}_N\}$ each with dimension of D. Each vector $\mathbf{x}_n \in \mathbf{X}$ is with a length of w (i.e., each vector has w points), the encoding procedure maps the vector to hypervector $\mathbf{Q}_n = \{q_{n,1}, q_{n,2}, ..., q_{n,D}\}$. This mapping is achieved by using a non-linear function and assigning a real value to each dimension of the hypervector based on the values in the corresponding dimension of the input vector $\mathbf{x}_n$. The encoding procedure uses the equation

$$q_{n,i} = \sin(\mathbf{x}_n \cdot \mathbf{G}_i^T) \times \cos(\mathbf{x}_n \cdot \mathbf{G}_i^T + b_i) \tag{1}$$

where each element of $\mathbf{G}_i$ follows $\mathcal{N}(0, 1)$ (i.e., normal distribution with mean of 0 and variance of 1). Here, b_i is randomly sampled from $[0, 2\pi]$ and follows a uniform distribution.

Finally, the information in each hypervector is summarized into various class hypervectors according to their label information. This is done by bundling (element-wise addition) the hypervectors in each class, resulting in a set of class hypervectors that represent the characteristics of each class. These class hypervectors can then be used to classify new instances by comparing their hypervector to the class hypervectors and assigning them to the closest class. The encoding procedure, therefore, provides a way to efficiently encode high-dimensional data and extract useful information for classification tasks in the hyperdimensional space. Since the dimension of hypervectors is significantly larger than the size of feature vectors, the quasi-orthogonality of the hypervectors can be guaranteed. For the different types of signal data, early fusion is employed where all signals are first synchronized using the same n-gram window, pooled together to create $\mathbf{x}_n$ vectors, and encoded using the encoding procedure described in Eq. (1).

3.2 Training

After encoded hypervectors are generated, a linear training element is integrated with each class (i.e., according to the label information in $\mathbf{Y}$) to investigate all-encompassing properties in data. Denote $\mathbf{C}_c = \{c_1, c_2, ..., c_D\}$ as the class hypervectors of class c and $\mathbf{Q}_{N+1}$ as the new encoded query data. Our model calculates the similarity between the incoming data $\mathbf{Q}_{N+1}$ and the class hypervector $\mathbf{C_c}$ to explore the closeness of two entries. The comparison can be formulated as

$$\delta(\mathbf{C_c}, \mathbf{Q}_{N+1}) = \frac{\mathbf{C_c} \cdot \mathbf{Q}_{N+1}}{\|\mathbf{C_c}\| \|\mathbf{Q}_{N+1}\|} \tag{2}$$

where $\cdot$ represents the inner product. If two hypervectors are similar to each other, the similarity value between them would be closer to 1. The proposed HDC framework integrates the encoded hypervectors by focusing on the new information appended to the class hypervectors. For example, HDC framework only considers a small portion to prevent model saturation if such information already exists in the hypervector. As a result, HDC provides the solution for efficient and accurate learning.

HDC offers iterative retraining to improve the framework performance. The retraining steps update the class hypervectors when incoming query data is mispredicted by the current model. For example, if the current HDC framework predicts a query c as c', the iterative learning step updates the class hypervectors as:

$$\begin{aligned} \mathbf{C}_c &\leftarrow \mathbf{C}_c + \eta(1 - \delta(\mathbf{C}_c, \mathbf{Q})) \times \mathbf{Q} \\ \mathbf{C}_{c'} &\leftarrow \mathbf{C}_{c'} - \eta(1 - \delta(\mathbf{C}_{c'}, \mathbf{Q})) \times \mathbf{Q} \end{aligned} \tag{3}$$

where η indicates the learning rate of the HDC model. According to Eq. (3), when $\delta(\mathbf{C}_c, \mathbf{Q}) \simeq \delta(\mathbf{C}_{c'}, \mathbf{Q})$, the model only updates a small portion since the similarity value would be close to 1. Otherwise, the retraining algorithm will update the model significantly when the prediction is far from the true value because the value of $\delta(\mathbf{C}_c, \mathbf{Q})$ is significantly smaller than $\delta(\mathbf{C}_{c'}, \mathbf{Q})$. In summary, the model starts with relatively high accuracy and will require fewer iterations until convergence.

4. EXPERIMENTAL DESIGN

A LASERTEC 65 DED hybrid CNC DMG machine was used for the fabrication of a designed part in Connecticut Center for Advanced Technology (CCAT). The design of the part is illustrated in Figure 3 a) and b). 1040 steel blocks of dimensions 76.2 mm x 76.2 mm x 76.2 mm were selected as the material for the experiment. In total, 18 parts were fabricated with the same machine settings with worn bits replaced based on machining experience. During the machining process, more than 91 process parameters such as power, torque, load, and current were collected by the DMG machine and stored for further analysis. While this DMG machine consists of both a LASERTEC 65 direct energy deposition and 5-axis CNC DMG system, the data acquisition is according to the CNC module. Particularly, process signals

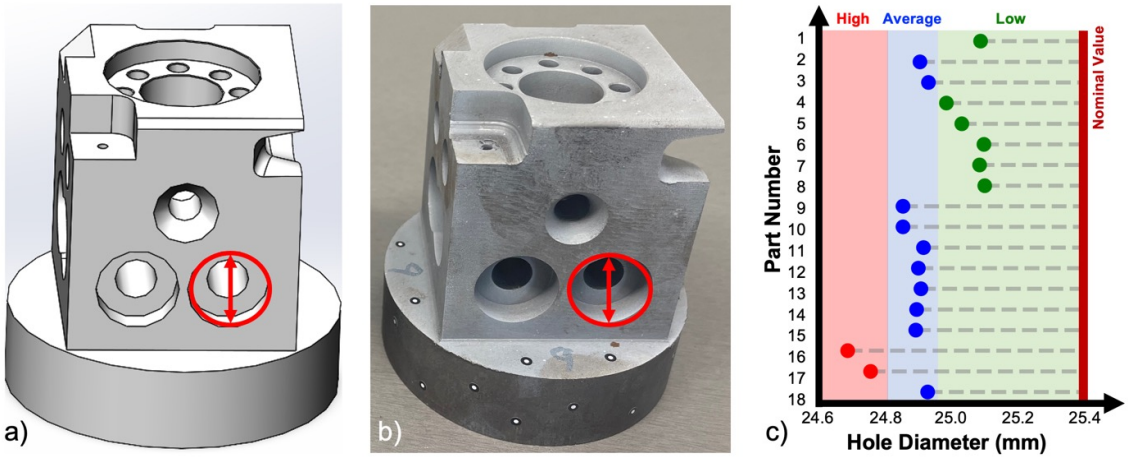

FIGURE 3: A) THE CAD MODEL OF THE MANUFACTURED PART. THE RED CIRCLE REPRESENTS THE FEATURE UTILIZED AS QUALITY CHARACTERIZATION. B) THE FINAL BUILD OF THE MANUFACTURED PART. C) MEASUREMENT, NOMINAL VALUE, AND DEVIATION OF FEATURE 20 (I.E., DIAMETER OF A 25.4 MM COUNTERBORE) FROM 18 DIFFERENT PARTS. ACTUAL MEASUREMENTS ARE SHOWN IN SCATTERS WITH THEIR CLASSES, NOMINAL VALUE FROM THE DESIGN IS INDICATED IN SOLID RED LINE, AND DEVIATIONS ARE PRESENTED IN GRAY DASHES.

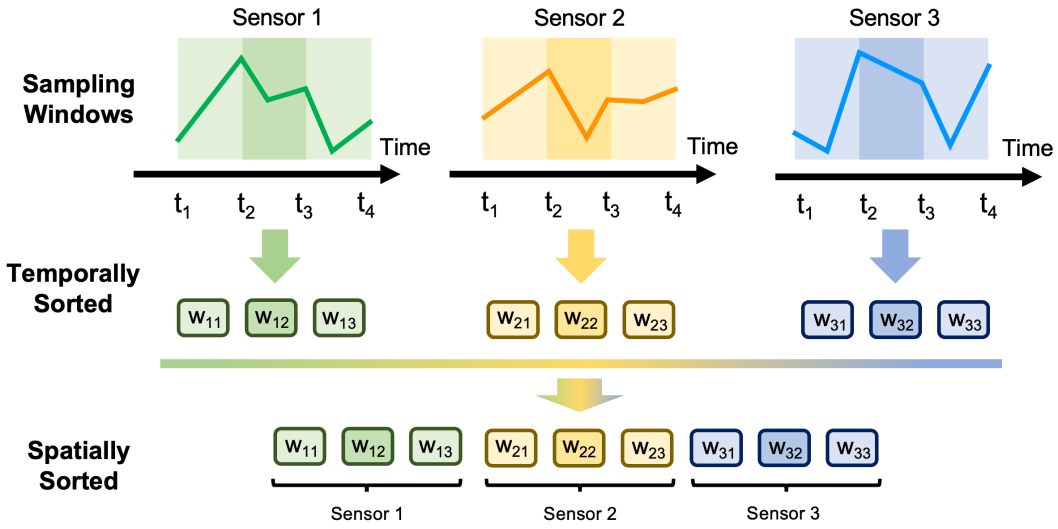

FIGURE 4: AN ILLUSTRATION OF THE SAMPLING PROCESS.

were recorded at 500 Hz using a Siemens Simatic IPC227E edge device. Signals such as torque, command speed, power, current, and axis position have six channels corresponding to the different 5 axes and spindle of the DMG machine. Signals were exported to labeled job files.

Each part includes 47 typical features such as chamfers, pockets, rounded corners, and through-holes. Tool path jobs were generated by Siemens NX and categorized into 42 different jobs according to the tooling bit needed. For example, jobs 03, and 04 were created using a 12.3 mm end mill bit while jobs 18, and 19 were created using a 6.76 mm drill bit. A job, or a command, can create multiple features. For instance, feature 31 (i.e., flatness) and feature 11 (i.e., profile) were both generated by job 01. A total of 41 in-process signal files for each of the 41 jobs were recorded for each part.

Post inspection of all 18 parts was conducted using a GOM ATOS ScanBox. All 47 features were measured and compared to its nominal value. Denoting measured feature value as y and the corresponded nominal value as y_n, the deviation of the feature i can be calculated as

$$d_{ij} = y_{ij} - y_{n_{ij}} \quad \forall i = 1, .., 47, j = 1, ..., 18 \tag{4}$$

where j represents the part number. Graphs were created for each feature to visualize the deviation from the nominal value and the feature with the most deviation was selected for further analysis.

5. EXPERIMENTAL RESULTS

Based on the deviation calculated from all 47 features from each part, feature 20 (i.e., diameter of a 25.4 mm counterbore) is selected to represent the quality deviation of the part. Figure 3 c) shows the measured feature of each part along with its deviation. The nominal value of the diameter is shown in a solid red line,

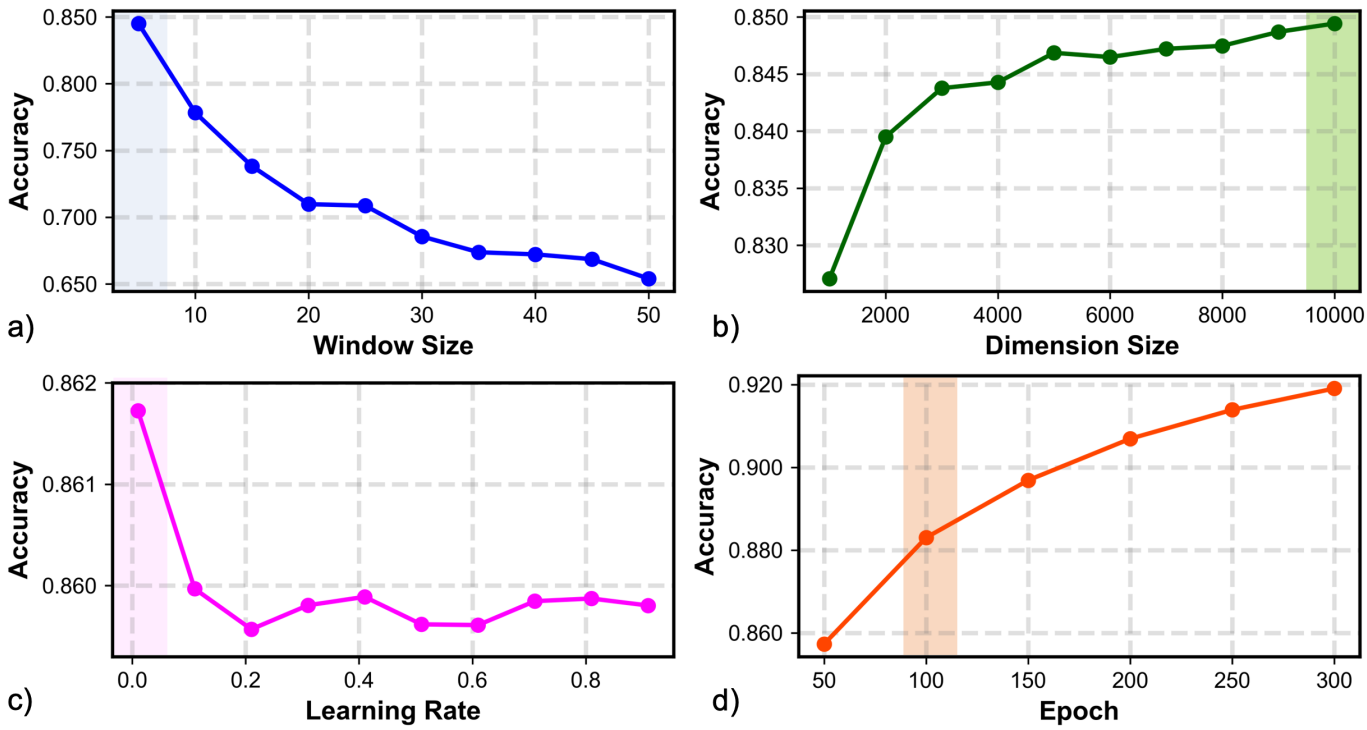

FIGURE 5: A) ACCURACY OF THE HDC BASED ON WINDOW SIZE IN THE RANGE OF 5 TO 50. THE WINDOW SIZE OF 5 PROVIDES THE HIGHEST RESULT AND IS SELECTED FOR FURTHER ANALYSIS. B) ACCURACY OF HDC BASED ON DIFFERENT SIZE DIMENSIONS D. THE HIGHLIGHTED BOX SHOWS THE SELECTED OPTIMAL DIMENSION D OF 10000 FOR FURTHER ANALYSIS. C) ACCURACY OF HDC ACCORDING TO LEARNING RATES VARYING FROM 0.1 TO 1. THE HIGHLIGHTED BOX SHOWS THE SELECTED OPTIMAL LEARNING RATE OF 0.01 FOR FURTHER ANALYSIS. D) ACCURACY OF THE HDC BASED ON THE NUMBER OF TRAINING EPOCHS FROM 50 TO 300. THE HIGHLIGHTED BOX SHOWS THE SELECTED OPTIMAL EPOCH NUMBER OF 100 FOR ANALYSIS.

actual measurements are shown in scatters with their classes, and deviations are presented in dashed gray lines. It can be observed that the deviation decreases significantly in part 11 in comparison with part 10. This is because of the tool replacement during the fabrication process; worn bits were replaced based on the observation and knowledge of the technician during the experiment. In addition, three classes (i.e., low deviation, average deviation, and high deviation) were created based on how each measured part deviated from the 25.4 mm nominal value.

Load, current, torque, command speed, control differential, power, and contour deviation were chosen as the input to the HDC model. Each of the signals had 6 channels and sampling was performed on all channels to generate enough data for training. Note that each channel was normalized before sampling. As shown in Figure 4, the sampling procedure uses n-gram windows moved without overlap over the time-series data. Each of these n-gram windows sampled during the same time interval are then concatenated together to create one feature vector sample. The objective of this sampling method is to generate more data samples and at the same time preserve the temporal relationship from the original signal. Testing was conducted on a range of n-gram windows from 5 to 50 to determine the optimal window size for further tuning. As shown in Figure 5 a), performance of the model drops significantly as window size increases. Therefore, the smallest n-gram window size of 5 was chosen as it provided the greatest accuracy. In the end, shuffling of the signals for each of the 18 parts was conducted to remove the impact from the order of production, and the number of samples from each class was balanced.

As the next step, the proposed HDC is introduced to fuse the preprocessed data to a high dimension. Particularly, Eq. (1) is utilized to fuse data from multiple sources into one single hypervector. Then, the tuning of the model parameters, i.e., dimension of hypervector D, learning rate η, and number of epochs k, was conducted to acquire the greatest accuracy for the final implementation of the HDC model. Note that both the tuning process and final implementation of the model are based on a 80% to 20% train-test split. We denote true positive, false positive, true negative, and false negative as TP, FP, TN, and FN respectively. The accuracy can be calculated as

$$\text{accuracy} = \frac{TP + TN}{TP + FP + TN + FN} \tag{5}$$

We first tune the dimensions D of the hypervectors using a range of dimensions from 1000 to 10000 with step sizes of 1000. As shown in Figure 5 b), the accuracy increases as the dimension D increases. The dimension that acquired the greatest accuracy (i.e., an accuracy of 84.9% at $D = 10000$) was selected for the tuning of both learning rate and epochs. Based on Figure 5 c), we then select the learning rate by implementing the encoding dimension of 10000. A range of values from 0.01 to 1 with step sizes of 0.1 was utilized for comparison. As shown, the model performance decreased with increasing learning rate albeit not

TABLE 1: MODEL PARAMETER SETTINGS AND MODEL STRUCTURE UTILIZED FOR COMPARISON.

Model	Parameters
SVM	Max Iterations = 100, Kernel Function = Radial basis function
LR	Max Iterations = 100, Solver = Limited-memory Broyden-Fletcher-Goldfarb-Shanno
NB	Additive Smoothing = 1.0
MLP	Max Iterations = 100, No. of Layers = 2, No. of Hidden Neurons = 100, Solver = Adam
ResNet	Max Iterations = 100, No. of Layers = 9, No. of Hidden Neurons = 128, Solver = Adam

significantly. The optimized learning rate was chosen when the model exhibited the greatest accuracy. This value occurred at a learning rate of 0.01 with an accuracy of 86.17%. Finally, the number of epochs is chosen based on the greatest increase in accuracy compared to the epoch tested before. Figure 5 d) depicts the accuracy based on the number of epochs from 50 to 300. An epoch of 100 was chosen as it provided the greatest increase in accuracy from a value of 85.7% to 88.3%.

After all parameters are tuned, the HDC model is compared with other conventional learning models such as support vector machines (SVM), logistic regression (LR), multinomial Naive-Bayes (NB), multilayer perceptron (MLP), and residual neural network (ResNet) to investigate the relationship between in-process signals and feature deviation. Parameters used for each learning model are summarized in Table 1. An 80% to 20% test-train split was again used and each algorithm was implemented 100 times and the average results were reported. The accuracy, precision, recall, and F1-score for each model is calculated as

$$\text{Precision} = \frac{TP}{TP + FP} \tag{6}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{7}$$

$$\text{F1} - \text{Score} = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \tag{8}$$

As shown in Table 2, the proposed HDC model, achieved an accuracy of 0.905, precision of 0.905, recall of 0.905, and F1-score of 0.905, which outperforms all other methods. Regarding accuracy, HDC performs 55.8%, 28.6%, 53.1%, 11.6%, and 28.0% better than SVM, LR, NB, MLP, and ResNet, respectively. The HDC model shows incredible robustness in its results as shown by the consistency across all performance metrics. Overall, the experimental results show that the proposed HDC framework provides superior results to other state-of-the-art methods and shows strong potential in characterizing multi-channel in-process signals to part deviation with limited samples of part data.

As the final step, we investigate the generalization of the proposed model on out-of-distribution data. Particularly, we utilized the first 80% of each signal's recording for training, and the last 20% is implemented for testing. Table 3 shows the performance of HDC and several ML models for performance of out-of-distribution data in terms of their accuracy, precision, recall, and F1-score. The HDC framework performs the best overall, with high values for accuracy, precision, recall, and F1-score. Particularly, the results according to the out-of-distribution data show

TABLE 2: PERFORMANCE METRICS (I.E., ACCURACY, PRECISION, RECALL, AND F1-SCORE) OF THE PROPOSED HDC MODEL COMPARED TO OTHER MACHINE LEARNING ALGORITHMS.

Model	Accuracy	Precision	Recall	F1-Score
HDC	0.905	0.905	0.905	0.905
SVM	0.347	0.370	0.347	0.325
LR	0.619	0.616	0.619	0.614
NB	0.374	0.522	0.374	0.341
MLP	0.789	0.806	0.789	0.786
ResNet	0.625	0.493	0.633	0.554

that the HDC archives accuracy, precision, recall, and F-score of 0.887, 0.896, 0.887, and 0.889, respectively. The metric results for HDC are similar to its metric values (i.e., 0.905) under the in-distribution condition that assumed a model should observe the data similar to training instances. The SVM model, on the other hand, performs poorly compared to the other models, with low values for accuracy, precision, recall, and F1-score. The LR and NB models perform moderately well. While LR performance is stable for different metrics, the NB model operates better at identifying true positives, but may also produce a higher number of false positives. The MLP model performs well, with high values for accuracy, precision, recall, and F1-score compared to ResNet for the out-of-distribution data. This suggests that the MLP with a lightweight structure and fewer hyperparameters is more flexible than ResNet to be generalized for out-of-distribution data with complex patterns in the data.

TABLE 3: PERFORMANCE METRICS (I.E., ACCURACY, PRECISION, RECALL, AND F1-SCORE) OF THE PROPOSED HDC MODEL COMPARED TO OTHER MACHINE LEARNING ALGORITHMS WITH OUT-OF-DISTRIBUTION DATA.

Model	Accuracy	Precision	Recall	F1-Score
HDC	0.887	0.896	0.887	0.889
SVM	0.343	0.368	0.343	0.322
LR	0.630	0.630	0.630	0.625
NB	0.420	0.600	0.420	0.384
MLP	0.797	0.818	0.797	0.795
ResNet	0.556	0.810	0.556	0.659

6. CONCLUSION

Despite the implementation of ML methods for in-process characterization, there are significant roadblocks that hinder the applications of these techniques, such as low sample efficiency, robustness to noise, and interpretability. Hyperdimensional computing (HDC) was introduced as a neurally inspired solution that

mimics the human brain to fuse multichannel in-process signals for the characterization of part deviation. Experimental results on a real-world hybrid 5-axis CNC Deckel-Maho-Gildemeister showed that HDC was able to fuse multichannel in-process signals (load, current, torque, command speed, control differential, power, and contour deviation) with 0.905 accuracy, precision, recall, and F1-score. Hyperdimensional computing has the potential to be implemented in various manufacturing processes for real-time process monitoring and quality control. For instance, HDC is capable of monitoring the printing process in real-time, detecting and correcting errors as they occur. By encoding data from sensors and cameras in high-dimensional vectors, HDC identifies deviations from expected patterns and alerts the operator to potential problems. This can help to prevent defects from occurring in the first place, rather than detecting them after the fact. In subtractive manufacturing settings, HDC is able to predict tool wear prediction by analyzing in-situ sensor data from the CNC machine. By encoding this data in high-dimensional vectors, hyperdimensional computing can identify patterns that indicate tool wear, such as changes in temperature or vibration. This can predict when the tool will need to be replaced or resharpened, allowing efficient use of the tool and reducing downtime.

ACKNOWLEDGMENTS

The authors gratefully acknowledge the valuable contributions from Connecticut Center for Advanced Technology (CCAT) for this research.

REFERENCES

[1] Imani, Farhad, Gaikwad, Aniruddha, Montazeri, Mohammad, Rao, Prahalada, Yang, Hui and Reutzel, Edward. "Layerwise in-process quality monitoring in laser powder bed fusion." *ASME 2018 13th International Manufacturing Science and Engineering Conference*. 2018. American Society of Mechanical Engineers Digital Collection.

[2] Imani, Farhad, Yao, Bing, Chen, Ruimin, Rao, Prahalad and Yang, Hui. "Joint multifractal and lacunarity analysis of image profiles for manufacturing quality control." *Journal of Manufacturing Science and Engineering* Vol. 141 No. 4 (2019).

[3] Chen, Ruimin, Reutzel, Edward W, Khanzadeh, Mojtaba and Imani, Farhad. "Heterogeneous quality characterization and modeling of thin wall structure in additive manufacturing." *Additive Manufacturing Letters* Vol. 3 (2022): p. 100042.

[4] Gibson, Ian, Rosen, David and Stucker, Brent. "Directed energy deposition processes." *Additive manufacturing technologies*. Springer (2015): pp. 245–268.

[5] Zhao, Xiyue, Imandoust, Aidin, Khanzadeh, Mojtaba, Imani, Farhad and Bian, Linkan. "Automated Anomaly Detection of Laser-Based Additive Manufacturing Using Melt Pool Sparse Representation and Unsupervised Learning." *2021 International Solid Freeform Fabrication Symposium*. 2021. University of Texas at Austin.

[6] Sefene, Eyob Messele, Hailu, Yeabsra Mekdim and Tsegaw, Assefa Asmare. "Metal hybrid additive manufacturing: state-of-the-art." *Progress in Additive Manufacturing* (2022): pp. 1–13.

[7] Popov, Vladimir V and Fleisher, Alexander. "Hybrid additive manufacturing of steels and alloys." *Manufacturing Review* Vol. 7 (2020): p. 6.

[8] Rescsanski, Sean, Imani, Mahdi and Imani, Farhad. "Heterogeneous Sensing and Bayesian Optimization for Smart Calibration in Additive Manufacturing Process." *ASME International Mechanical Engineering Congress and Exposition*, Vol. 86649: p. V02BT02A051. 2022. American Society of Mechanical Engineers.

[9] Merklein, Marion, Junker, Daniel, Schaub, Adam and Neubauer, Franziska. "Hybrid additive manufacturing technologies–an analysis regarding potentials and applications." *Physics procedia* Vol. 83 (2016): pp. 549–559.

[10] Zhou, Huicheng, Lang, Minglang, Hu, Pengcheng, Su, Zhiwei and Chen, Jihong. "The modeling, analysis, and application of the in-process machining data for CNC machining." *The International Journal of Advanced Manufacturing Technology* Vol. 102 No. 5 (2019): pp. 1051–1066.

[11] Qi, Qinglin and Tao, Fei. "Digital twin and big data towards smart manufacturing and industry 4.0: 360 degree comparison." *Ieee Access* Vol. 6 (2018): pp. 3585–3593.

[12] Brillinger, Markus, Wuwer, Marcel, Hadi, Muaaz Abdul and Haas, Franz. "Energy prediction for CNC machining with machine learning." *CIRP Journal of Manufacturing Science and Technology* Vol. 35 (2021): pp. 715–723.

[13] Wang, Dashuang, Hong, Rongjing and Lin, Xiaochuan. "A method for predicting hobbing tool wear based on CNC real-time monitoring data and deep learning." *Precision Engineering* Vol. 72 (2021): pp. 847–857.

[14] Patange, Abhishek D and Jegadeeshwaran, R. "Application of bayesian family classifiers for cutting tool inserts health monitoring on CNC milling." *International Journal of Prognostics and Health Management* Vol. 11 No. 2 (2020).

[15] Ge, Lulu and Parhi, Keshab K. "Classification using hyperdimensional computing: A review." *IEEE Circuits and Systems Magazine* Vol. 20 No. 2 (2020): pp. 30–47.

[16] Zou, Zhuowen, Alimohamadi, Haleh, Zakeri, Ali, Imani, Farhad, Kim, Yeseong, Najafi, M Hassan and Imani, Mohsen. "Memory-inspired spiking hyperdimensional network for robust online learning." *Scientific reports* Vol. 12 No. 1 (2022): pp. 1–13.

[17] Chen, Ruimin, Imani, Mohsen and Imani, Farhad. "Joint active search and neuromorphic computing for efficient data exploitation and monitoring in additive manufacturing." *Journal of manufacturing processes* Vol. 71 (2021): pp. 743–752.

[18] Chen, Ruimin, Sodhi, Manbir, Imani, Mohsen, Khanzadeh, Mojtaba, Yadollahi, Aref and Imani, Farhad. "Brain-inspired computing for in-process melt pool characterization in additive manufacturing." *CIRP Journal of Manufacturing Science and Technology* Vol. 41 (2023): pp. 380–390.

[19] Zou, Zhuowen, Kim, Yeseong, Imani, Farhad, Alimohamadi, Haleh, Cammarota, Rosario and Imani, Mohsen.

"Scalable edge-based hyperdimensional learning system with brain-like neural adaptation." *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*: pp. 1–15. 2021.

[20] Imani, Farhad and Chen, Ruimin. "Latent Representation and Characterization of Scanning Strategy on Laser Powder Bed Fusion Additive Manufacturing." *ASME International Mechanical Engineering Congress and Exposition*, Vol. 86649: p. V02BT02A009. 2022. American Society of Mechanical Engineers.

[21] Poduval, Prathyush, Alimohamadi, Haleh, Zakeri, Ali, Imani, Farhad, Najafi, M Hassan, Givargis, Tony and Imani, Mohsen. "Graphd: Graph-based hyperdimensional memorization for brain-like cognitive learning." *Frontiers in Neuroscience* Vol. 16 (2022): p. 5.

[22] Imani, Mohsen, Zakeri, Ali, Chen, Hanning, Kim, Tae-Hyun, Poduval, Prathyush, Lee, Hyunsei, Kim, Yeseong, Sadredini, Elaheh and Imani, Farhad. "Neural computation for robust and holographic face detection." *Proceedings of the 59th ACM/IEEE Design Automation Conference*: pp. 31–36. 2022.

[23] Poduval, Prathyush, Issa, Mariam, Imani, Farhad, Zhuo, Cheng, Yin, Xunzhao, Najafi, Hassan and Imani, Mohsen. "Robust In-Memory Computing with Hyperdimensional Stochastic Representation." *2021 IEEE/ACM International Symposium on Nanoscale Architectures (NANOARCH)*: pp. 1–6. 2021. IEEE.

[24] Tangjitsitcharoen, Somkiat. "In-process monitoring and detection of chip formation and chatter for CNC turning." *Journal of Materials Processing Technology* Vol. 209 No. 10 (2009): pp. 4682–4688.

[25] Lamraoui, MMEF, El Badaoui, M and Guillet, F. "Chatter detection in CNC milling processes based on Wiener-SVM approach and using only motor current signals." *Vibration engineering and technology of machinery*. Springer (2015): pp. 567–578.

[26] Zhang, Chun Liang, Yue, Xia, Jiang, Yong Tao and Zheng, Wei. "A hybrid approach of ANN and HMM for cutting chatter monitoring." *Advanced Materials Research*, Vol. 97: pp. 3225–3232. 2010. Trans Tech Publ.

[27] Sun, Huibin, Zhang, Xianzhi and Wang, Junyang. "Online machining chatter forecast based on improved local mean decomposition." *The International Journal of Advanced Manufacturing Technology* Vol. 84 No. 5 (2016): pp. 1045–1056.

[28] Chungchoo, C and Saini, D. "On-line tool wear estimation in CNC turning operations using fuzzy neural network model." *International Journal of Machine Tools and Manufacture* Vol. 42 No. 1 (2002): pp. 29–40.

[29] Martínez-Arellano, Giovanna, Terrazas, German and Ratchev, Svetan. "Tool wear classification using time series imaging and deep learning." *The International Journal of Advanced Manufacturing Technology* Vol. 104 No. 9 (2019): pp. 3647–3662.

[30] He, Zhaopeng, Shi, Tielin, Xuan, Jianping and Li, Tianxiang. "Research on tool wear prediction based on temperature signals and deep learning." *Wear* Vol. 478 (2021): p. 203902.

[31] Cheng, Minghui, Jiao, Li, Yan, Pei, Jiang, Hongsen, Wang, Ruibin, Qiu, Tianyang and Wang, Xibin. "Intelligent tool wear monitoring and multi-step prediction based on deep learning model." *Journal of Manufacturing Systems* Vol. 62 (2022): pp. 286–300.

[32] Zhang, Nian and Shetty, Devdas. "An effective LS-SVM-based approach for surface roughness prediction in machined surfaces." *Neurocomputing* Vol. 198 (2016): pp. 35–39.

[33] Wang, Xingsheng, Kang, Min, Fu, Xiuqing and Li, Chunlin. "Predictive modeling of surface roughness in lenses precision turning using regression and support vector machines." *The International Journal of Advanced Manufacturing Technology* Vol. 87 No. 5 (2016): pp. 1273–1281.

[34] Do Duc, T, Ba, N Nguyen, Van, C Nguyen, Nhu, T Nguyen and Tien, D Hoang. "Surface roughness prediction in CNC hole turning of 3X13 steel using support vector machine algorithm." *Tribology in Industry* Vol. 42 No. 4 (2020): p. 597.

[35] Xie, Nan, Zhou, Junfeng and Zheng, Beirong. "An energy-based modeling and prediction approach for surface roughness in turning." *The International Journal of Advanced Manufacturing Technology* Vol. 96 No. 5 (2018): pp. 2293–2306.

[36] Wu, TY and Lei, KW. "Prediction of surface roughness in milling process using vibration signal analysis and artificial neural network." *The International Journal of Advanced Manufacturing Technology* Vol. 102 No. 1 (2019): pp. 305–314.

[37] Lin, Wan-Ju, Lo, Shih-Hsuan, Young, Hong-Tsu and Hung, Che-Lun. "Evaluation of deep learning neural networks for surface roughness prediction using vibration signal analysis." *Applied Sciences* Vol. 9 No. 7 (2019): p. 1462.