

CIRP Conference on Manufacturing Systems (CIRP CMS)

3D drill hole geometry prediction from force and torque data

 Ole Stüven^{a,*},  Keno Moenck^a,  Malte Flehmke^b,  Jan Dege^b,  Thorsten Schüppstuhl^a^aHamburg University of Technology, Institute of Aircraft Production Technology, Denickestraße 17, 21073 Hamburg, Germany^bHamburg University of Technology, Institute of Production Management and Technology, Denickestraße 17, 21073 Hamburg, Germany* Corresponding author. E-mail address: ole.stueven@tuhh.de**Abstract**

Drilling is one of the most frequently used manufacturing methods. In aircraft manufacturing, common passenger planes require a very large number of drill holes. Verifying whether these holes meet specifications is a labor-intensive task that could benefit from automation. However, due to the large number of holes, in-process measurement is time-consuming and can delay production, rendering it infeasible. An alternative approach is to predict hole quality using process data collected during drilling. Previous studies have demonstrated the feasibility of extracting valuable information about manufacturing processes, tools, and products from such data. However, only single features of the drill holes are predicted. In this work, we propose a method to predict the hole geometry based on force and torque time series data recorded during the drilling process. The resulting point cloud can then be used to predict multiple features or serve as the basis for a more advanced quality control process that is not solely based on verifying whether a few geometric parameters fall within specification.

© 2026 The Authors. Published by Elsevier B.V.

This is an open access article under the CC BY-NC-ND license (<https://creativecommons.org/licenses/by-nc-nd/4.0>)

Peer-review under responsibility of the scientific committee of the 2026 CIRP CMS

Keywords: manufacturing; drilling; hole quality prediction; deep learning**1. Introduction**

Depending on the usage of a drill hole, it has to fulfill certain specifications. Currently, drill holes are measured sample-based in production. This poses the risk of missing holes that are not within specifications. This is especially disadvantageous, as drilling processes often produce a coherent set of bad holes. The earlier this is detected, the less rework has to be done, saving cost and time. In-process hole quality control can minimize this risk. There are two fundamental approaches to this task: direct and indirect measurement. For direct measurement, hole quality is measured with the respective measuring device. For indirect measurement, hole quality is inferred from data measured during drilling, e.g., torque, force, vibration, and acoustic emission. The key advantage of direct measurement is that it can be accepted as ground truth. The key disadvantage is that it increases production time, which also increases production cost, and therefore renders it infeasible in most cases. Indirect measurement is a more cost-effective method, but it requires sophisticated models that can predict hole quality based on process data. Often, a data-based approach is used. Therefore, training data has to be recorded in a separate phase prior to production

[1]. Current methods extract features from the time series data and then use a machine learning regressor or classifier to predict the quality of the holes [2, 3, 4, 5, 6]. In recent years, Deep Learning (DL)-based methods have emerged in this research field with promising results. Compared to classical feature extraction, DL models learn the features they extract, which can deliver superior results [7]. Usually, these models predict a single value, for example, delamination or surface roughness. This is reasonable since the quality control process is based on verifying whether certain directly measurable quality features are within the specified limits. However, this work explores one step further. The idea is to predict the hole geometry instead of single values. This enables the direct measurement of various geometric quality features within the predicted geometry, rather than training a separate model for each feature. As an example, one can apply virtual gauging in the predicted geometry [8]. Since conventional gauging is often used in production, this could lower the entry barrier to automation. Furthermore, this paves the way for a more advanced quality control process that is not based on single values, but a holistic analysis of the holes' geometries. Finally, having a full geometric representation of the model's prediction can help improve its explainability to the end user.

This work contributes three-folded:

- We propose a method to predict the 3D geometry of drill holes from time series process information.
- We develop a model architecture that is able to perform this task, as well as an ablation study to prove its effectiveness.
- We generated a dataset and conducted experiments on real-world data that empirically prove our method.

The rest of this work is structured as follows: First, in Section 2, we outline existing applications that use DL to predict hole quality from process data. Next, we describe our proposed method in Section 3. In Section 4, we describe our experiments and present the results in Section 5. Finally, we discuss and conclude the findings in Section 6.

2. Related Works

Recent research in the field of hole quality prediction based on process data with DL models focuses on the prediction of single values. In [9], the authors use 1D Convolutional Neural Networks (CNN) to predict the delamination factor for holes drilled in CFRP by an industrial robot. Each sensor signal is passed through a 1D CNN with 3 convolutions, a max pooling layer, and a dense layer. The features are concatenated, and a final dense layer predicts the output value. In [10], drilling in CFRP is examined. The authors utilize Stacked Sparse Autoencoders (SSAEs) to extract deep features from the time and frequency features of different sensor signals. These deep features are then fed into a Bayesian Long Short-Term Memory (LSTM) to predict the statistical delamination Factor. In [11], the authors aim to predict surface roughness of drill holes made by CNC machines. They convert the sensor signals via Gramian Angular Field (GAF) [12] and Short-Time Fourier Transform (STFT), which are then processed by 2D convolution. GAF represents time series as a 2D image by converting the series to polar coordinates and computing the cosine or sine of the sum or difference of the corresponding angles, using the x and y coordinates of the image as indices of the time series. The resulting features are fused and processed by a Representative Vision Transformer (RepViT) [13] to predict the surface roughness. RepViT combines CNNs with transformers and reparameterizes the model after training to increase efficiency. In [14], a model is developed, that is able to generate images of drill holes in CFRP from process data. The time series are first converted to GAF, which are then fed into a Generative Adversarial Network (GAN)-based image-to-image model, converting the GAF image into an image of the drill hole.

3. Method

In this section, first, the basic principle for predicting 3D geometry of drill holes from process data is described. Next, the considerations leading to the DL model that is at the core of the

method is developed. Afterwards, the whole pipeline, consisting of preprocessing and the model itself is proposed. Finally, the loss used for training is presented.

3.1. Basic principle

Given the complexity of the problem, DL is the most promising approach due to its ability to capture deep correlations between the input and output data. Since a drill hole closely resembles a cylinder, it can be unwrapped along its center axis and rasterized to a 2D grid with minimal information loss, reducing the complexity of the problem. The only exception are undercuts that intersect the same line, that is perpendicular to the center axis, more than once. Since this is highly unlikely, this exception is neglected. Therefore, we can use a model that is able to predict a 2D image from a time series.

3.2. Time series to image models

Theoretically, the method shown in [11] is a viable approach to the problem. However, it has two disadvantages. Firstly, the input time series is scaled down significantly to create the GAF. In the paper, a GAF of size 256×256 is used. Due to the principle of the GAF transformation, this is equivalent to an effective time series length of 256. The time series for this problem has a length of about 30,000. This would mean a reduction factor of more than 100. This means that high-frequency information is lost, which may be important for the prediction. Furthermore, the paper aims to produce images that are perceived as authentic by humans, while, for the task at hand, it is essential that the values of each pixel in the image are as close to the ground truth as possible. Therefore, we chose a different approach. The time series is fed into a network designed for time series regression. The last layer of these networks is usually a dense layer that predicts a single value from the features, which can be modified to output a 2D image. This image can then be transformed into a 3D point cloud, representing the geometry of the drill hole. Since there is no existing research on the task at hand, DL architectures used for similar tasks in related research serve as a basis. However, using transformations such as GAF or STFT before feeding the data into the model is not considered, as these transformations usually cause a loss of information. Since the intended output is more complex than the usual single value that the other models output, the idea is to feed all the information into the model and let it learn to extract the important features. 1D CNNs have been proven to be effective in general time series classification research [15] and also in recent related research [9]. Therefore, as a baseline, a model based on 1D convolutions is chosen. To transform the deep features into an image, a fully connected layer is used to create a 1D vector that can be rearranged into a rectangular grid, an image.

3.3. Data preprocessing

The only preprocessing step applied to the time series is aligning the start of the drilling process for each individual hole and cropping each sequence to a fixed length. The alignment

point is determined by identifying the first timestamp where the raw force sensor signal exceeds a fixed threshold, as shown in Figure 1.

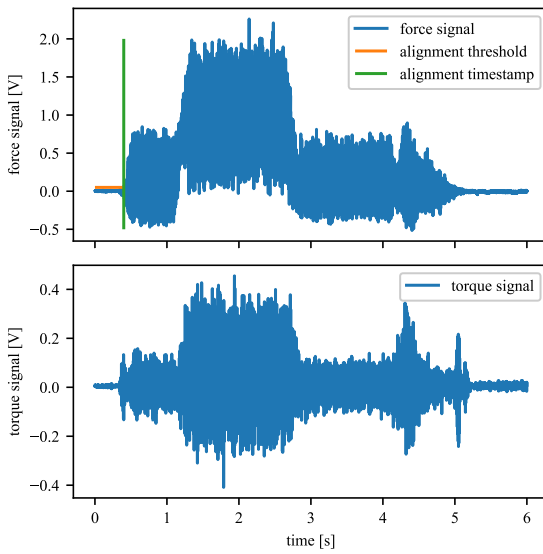


Fig. 1. Signals of the sensor data during drilling: Force in drilling direction on the top and torque around drilling direction on the bottom. The threshold values indicate the method for aligning the drill data.

Instead of cropping the signal to the main interval where the drill bit is in contact with the workpiece, an interval before and after the main interval is kept. This is done intentionally to detect, for example, if chips are stuck to the drill bit or if the drill bit is damaged. Therefore, a length of 30,000 and a threshold of 0.05 V is chosen. These values have to be adjusted for different processes. As input, we use the sensors' output voltages. We do not need to convert these to the corresponding SI-units, i.e., Nm or N, since it is only a linear scaling.

The point clouds of the holes are transformed into 2D images of size 14×28 . The pipeline is depicted in Figure 2. Firstly, the center axis of the hole is determined. Since the point cloud is distributed evenly around the center axis and the holes are drilled with minimal angular deviation, the mean of the measured points defines the position of the axis, and the direction of the axis is assumed to be parallel to the z-axis. The Cartesian coordinates are then transformed to cylindrical coordinates with respect to the determined center axis. The angle is mapped to the x-axis of the image, and the z-value is mapped to the y-axis of the image. The radius is min-max-scaled to be in the range $[0, 1]$ and corresponds to each pixel in the image. The min and max values are taken from the total set of images. The point cloud is measured with a very high resolution around the circumference, however, the image has a resolution of 28 in the corresponding direction. Therefore, the average value of the corresponding values is calculated for each pixel.

3.4. Model architecture

The architecture is based around the encoder-decoder nature of the problem. In this section, the encoder is described.

It consists of 6 1D CNN layers. Each CNN layer consists of a 1D convolution, a 1D batch normalization, a ReLU as the activation function, and spatial dropout. The 1D convolution has a kernel size of 9, a stride of 2, and no padding. The in and out channels are each a consecutive pair in the sequence $[2, 4, 8, 16, 32, 64, 128]$. We chose to use multiple layers to enable the model to extract deep features from the time series. The doubling of the channels in each layer, paired with the stride of 2, keeps the amount of information roughly equal during the convolution, which is found to be favorable for the problem. To follow the encoder-decoder architecture pattern, we introduce a bottleneck; an average pooling is applied after the convolution, which prevents the model from overfitting. The spatial dropout masks whole channels from the convolution result.

3.5. Loss

Since the orientation around the center axis of the drill hole is irrelevant for determining its quality, the model should not need to predict this orientation. Therefore, a rolling loss function is used that computes the Mean Squared Error (MSE) across all pixels for each shift in the y-direction of the image and then chooses the minimal loss from them. The formula for the MSE_i used for a single image is shown in Equation 1. The formula for a set of N images is shown in Equation 2.

$$MSE_i = \min_{0 \leq r \leq K-1} \frac{1}{J \cdot K} \left(\sum_{j=1}^J \sum_{k=1}^K (y_{i,j,k} - \hat{y}_{i,j,k+r})^2 \right) \quad (1)$$

$$MSE = \frac{1}{n} \sum_{i=1}^n MSE_i \quad (2)$$

Here, J and K are the height and width dimensions of the image and j and k the corresponding indices. r is the rolling index, y is the ground truth image and $\hat{y}$ is the prediction. If $k + r > K$, K is subtracted from the resulting index. The use of this loss does not affect the interpretation of the error metrics, it only removes the ability of the model to predict the rotation of the holes.

4. Experiments

This section is divided into the description of the drilling experiment, the training of the DL models, and an ablation study.

4.1. Drilling experiment

The drilling was done on a Seti-Tec M-EDU that is guided by a CNC machine to simulate the conditions in aircraft manufacturing. The whole experimental setup is depicted in Figure 3(a). The Electric Drilling Unit (EDU) was set to an RPM of $4,000 \text{ min}^{-1}$, a feed of 0.1 mm using Peck-Feed with 2.5 strokes per revolution and an amplitude of 0.15 mm and minimum quantity lubrication. The drill has a nominal diameter of 6.26 mm with a countersink. The test plate is made of 10 mm thick Al2024, which is shown in Figure 3(b). The holes

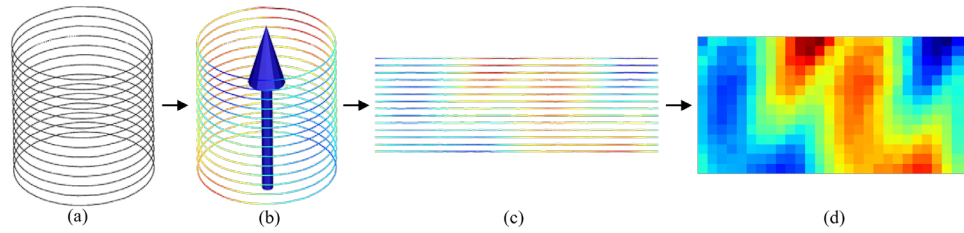


Fig. 2. Unwrapping of the point cloud (the colormap represents the radius of each point): (a) point cloud as measured by CMM; (b) center axis in point cloud; (c) unwrapped point cloud; (d) 2D image representation of the hole geometry.

were drilled in a 27×27 grid with a distance of 10 mm in each direction. Force in z-direction and torque around the z-axis were measured during drilling by a Kistler 4-Component Dynamometer Type 9272 with a sample rate of 5,000 Hz. Due to some errors during data storage, we can only use 690 of the 729 holes for analysis. We measured the plate using a Coordinate Measuring Machine (CMM), a Mitutoyo Crysta-Apex S574. The resulting point cloud per hole is depicted in Figure 2(a). For simplicity, the countersink of the hole was not measured. The lateral resolution is very high, as this does not require more measuring time. However, the vertical resolution can only be increased with more scans, which scales the measuring time linearly. Therefore, 14 rings were chosen for the vertical resolution, enough to prove that the method works.

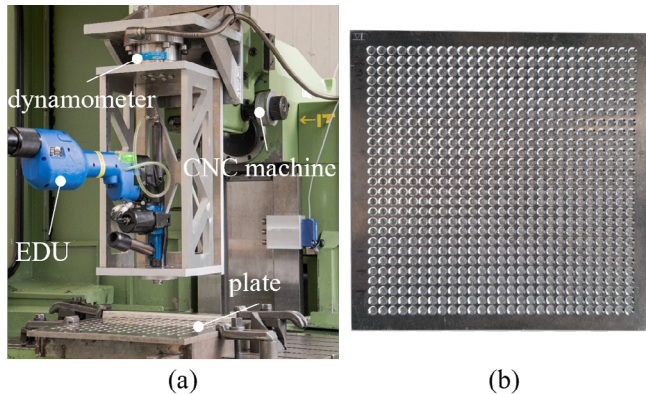


Fig. 3. (a) Experimental setup. (b) Top view of the aluminum plate with the drilled holes.

4.2. Model training

The training was conducted on an NVIDIA RTX 3090 GPU. The dataset is split at random into 5 sets of train, validation, and test data at ratios of 0.7, 0.15, and 0.15, respectively. The models are trained for 3,000 epochs on each split. The validation data was used to determine the optimal epoch, where the test loss and Coefficient of Determination (R^2) are evaluated. The formula for R^2 is shown in Equation 3.

$$R^2 = 1 - \frac{RSS}{TSS} \quad (3)$$

Here, RSS is the residual sum of squares, which is equal to the $MSE \cdot n$. TSS is the total sum of squares, where the difference from each y_i to the mean $\bar{y}$ is squared and then summed. The challenge was to determine an appropriate mean $\bar{y}$, because the MSE uses the rolling loss function. Simply taking the mean of all images as they are would not be consistent with the rolling loss. Therefore, the mean that yields the minimal TSS was determined using a model, that fits a single image to the test data images until it converges, using the rolling loss function. Opposed to the MSE , the R^2 metric shows the performance of the model in relation to the deviation within the test set. The models are implemented in PyTorch [16]. We chose Adam [17] as an optimizer with a learning rate of 0.001 and a batch size of 64.

4.3. Ablation study

An ablation study is performed, in which the number of layers, kernel size, and activation function are varied, as these are the major hyperparameters of the model. The numeric hyperparameters are tested in the direct vicinity of the base model. For the activation function, GELU and no activation function are tested. Table 1 lists all the hyperparameter variations.

Table 1. Hyperparameter variations of the model.

Id	Convolution channels	Kernel size	Activation
V0	[2, 4, 8, 16, 32, 64, 128]	9	ReLU
V1	[2, 4, 8, 16, 32, 64]	9	ReLU
V2	[2, 4, 8, 16, 32, 64, 128, 256]	9	ReLU
V3	[2, 4, 8, 16, 32, 64, 128]	7	ReLU
V4	[2, 4, 8, 16, 32, 64, 128]	11	ReLU
V5	[2, 4, 8, 16, 32, 64, 128]	9	GELU
V6	[2, 4, 8, 16, 32, 64, 128]	9	None

5. Result

First, the results of the ablation study are shown. For each model variation and split, the MSE , multiplied by 1,000 for readability, is listed in Table 2. R^2 , multiplied by 100 for readability, is listed in Table 3. Also, the mean and standard deviation for each model variation are listed. The best result is

marked in bold. The results confirm that, on average, the base model is the best-performing configuration. Neither increasing nor decreasing the number of layers nor the kernel size yields a better result. The same is true for changing the activation from ReLU to GELU or no activation.

Table 2. MSE ($\times 1,000$) for each split and variation of the model.

split	V0	V1	V2	V3	V4	V5	V6
0	1.58	1.49	1.43	1.65	2.29	2.16	2.64
1	1.17	1.30	1.18	1.27	1.25	1.21	2.46
2	1.22	1.72	1.31	1.22	1.20	1.51	2.09
3	1.68	1.75	1.68	1.66	2.35	1.58	3.05
4	1.16	1.11	1.76	1.23	1.91	1.15	2.18
mean	1.36	1.47	1.47	1.41	1.80	1.52	2.48
std	0.25	0.27	0.25	0.23	0.55	0.40	0.39

Table 3. R^2 ($\times 100$) for each split and variation of the model.

split	V0	V1	V2	V3	V4	V5	V6
0	42.7	46.0	48.1	40.3	17.1	21.9	4.3
1	56.2	51.3	55.7	52.2	53.2	54.5	7.5
2	49.1	28.1	45.4	49.0	49.7	36.9	12.8
3	50.0	48.2	50.1	50.6	30.2	53.1	9.6
4	48.0	50.1	20.9	44.9	14.5	48.6	2.3
mean	49.2	44.7	44.0	47.4	32.9	43.0	7.3
std	4.8	9.5	13.5	4.8	17.9	13.7	4.1

As the images are scaled to a range of $[0, 1]$ for training, the MSE of 0.00136 corresponds to a RMSE of $12.5\mu\text{m}$ in real scale. This corresponds to an error of 0.4% in relation to the nominal radius. Figure 4 depicts a histogram of the error of a single hole that has representative error metrics for the model. The 10th percentile is at $13.5\mu\text{m}$ and the 90th percentile at $18.3\mu\text{m}$. The predicted image, along with the ground truth from the same hole, is depicted in Figure 5. Depending on the application, the image can be transformed back into a 3D point cloud, although with lower lateral resolution than the original point cloud. From the images we can derive that the model is able to capture the characteristic structure of the holes, especially of the three grooves.

To test whether we can derive conventional quality features from the predicted point cloud, each of the 14 diameters was calculated from the predicted image and compared with those from the ground-truth images. The R^2 for a split with representative results for the model are depicted in Figure 6. The score almost constantly reduces from top to bottom.

To examine how the data size affects the performance of the model, the train size ratio is varied in $[0.5, 0.6, 0.7, 0.8, 0.9]$. The test and validation ratio are set to half of the remaining size

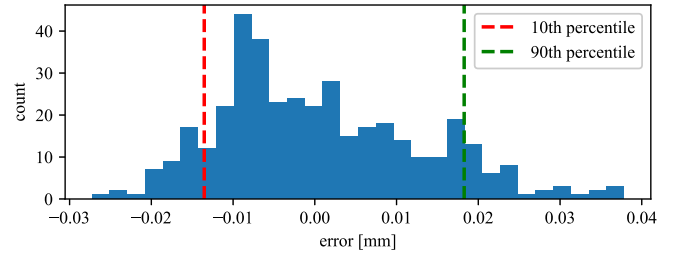


Fig. 4. Histogram of the error for a hole with representative error metrics.

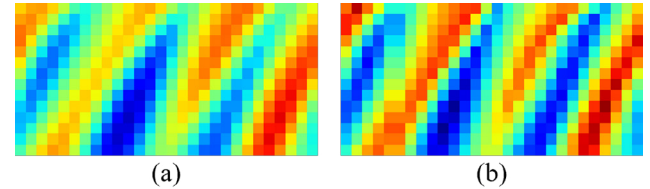


Fig. 5. (a) prediction of the model for a hole with representative MSE . (b) ground truth for that hole.

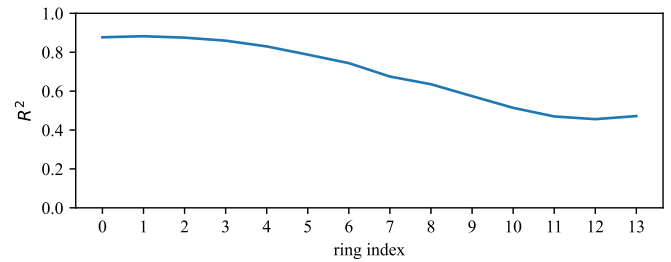


Fig. 6. R^2 for each diameter (ring index) in the predicted images for a representative split of the model.

each. For each ratio, the chosen model is trained with 5 random splits. The results are listed in Table 4.

Table 4. Mean of MSE ($\times 1,000$) and R^2 ($\times 100$) for each train size ratio over 5 random splits.

train size ratio	0.5	0.6	0.7	0.8	0.9
MSE	1.43	1.37	1.36	1.40	1.48
R^2	45.3	48.7	49.2	45.7	47.0

6. Discussion and Conclusion

The model reached an $RMSE$ of $12.5\mu\text{m}$ and R^2 of 0.492. The $RMSE$ is small, but it depends on the use case, if the $RMSE$ is sufficient. The R^2 shows, that the model is able to explain roughly 50% of the variance in the test data. This shows, that the model is able to predict 3D geometry of drill holes from just two channels of force and torque data recorded during drilling. The results of the ablation study prove, that the pro-

posed model architecture is the most effective within the vicinity of architectures with varied hyperparameters. Qualitatively, the results show that the model is able to capture and accurately predict the characteristic structure of the hole geometry. The dataset is only taken from a single plate. Therefore, all experimental parameters regarding the machine and material are the same. The currently trained model would probably perform poorly on new data that is recorded under different experimental parameters. However, if the model is trained with a variety of parameters, it might be able to generalize to new data, recorded with unseen parameters. Also, for real use cases, it might not be necessary for a model to be able to generalize over a huge spectrum of parameters, as a model could be trained for a specific process.

The decrease of the R^2 score, as the derived diameter is lower in the hole might be due to the data being more informative on the top section of the hole, or the model being more sensitive to information concerning the top section of the hole. As shown in Table 4, the performance of the model is roughly equal under varying train size ratios, meaning less data might be needed to train the model. Furthermore, there are only two data channels captured. The results might be improved by recording more channels, especially in the lateral plane, as this is the plane, in which the radius lies.

To the best of our knowledge, we are the first to propose a method for 3D hole geometry prediction from time series data. There are two main differences to established indirect inspection methods. Firstly, they rely more on statistical feature extraction, and secondly, they only predict single values. The use of DL in our approach instead of feature extraction has the potential to extract more information out of the data. However, generally, more training data and effort for model design and training is necessary with DL approaches. It depends on the use case, if this is viable. The results representing 3D hole geometry can be used for advanced evaluation of the hole, like virtual gauging, deriving single geometrical features or improving the explainability of predictions by providing a visual representation of the predicted drill hole. Here, the viability also depends on the use case. However, with big industries like aircraft manufacturing increasing automation, novel measurement methods like ours could support this process. The presented possibilities can be explored in future research. Furthermore, the limitations can be addressed by using bigger datasets with more variety. In addition, further model architectures can be explored to improve the predictions.

Acknowledgements

The research leading to these results received funding from the *Federal Ministry for Economic Affairs and Energy (BMWE)* in the project *HoleListic* under the Grant Number 20D2205C as part of the *Federal Aeronautical Research Programme LuFo VI-3*.

Supported by:



on the basis of a decision by the German Bundestag

References

- [1] O. Stüven, F. Geiger, K. Moenck, T. Schüppstuhl, Towards Explainable AI-based In-line Inspection and Quality Assurance for Drilling in Aircraft Manufacturing, in: *Annals of Scientific Society for Assembly, Handling and Industrial Robotics 2024*, Springer International Publishing, status: in press.
- [2] V. Mathiyazhagan, A. Meena, Predictive modelling of tool wear in CFRP drilling using acoustic emission sensors under dry and cryogenic conditions, *Wear* 570 (2025) 205930. doi:10.1016/j.wear.2025.205930.
- [3] P. Piankitrunreang, K. Chaiprabha, W. Chungsangsatiporn, C. Ratana-sumawong, P. Chancharoen, R. Chancharoen, Acoustic-Based Machine Main State Monitoring for High-Speed CNC Drilling, *Machines* 13 (5) (2025) 372. doi:10.3390/machines13050372.
- [4] M. Flehmke, D. Romanenko, O. Rosenthal, J. Hendrik Dege, Klassifizierung von Prozesszuständen beim Bohren, *Zeitschrift für wirtschaftlichen Fabrikbetrieb* 118 (5) (2023) 302–306. doi:10.1515/zwf-2023-1054.
- [5] C. Zhang, X. Song, Z. Liu, B. Ma, Z. Lv, Y. Su, G. Li, Z. Zhu, Real-time and multi-objective optimization of rate-of-penetration using machine learning methods, *Geoenergy Science and Engineering* 223 (2023) 211568. doi:10.1016/j.geoen.2023.211568.
- [6] J. Koch, D. Schoepflin, A. Venkatanarasimhan, T. Schüppstuhl, Tool Wear Classification in Automated Drilling Operations of Aircraft Structure Components using Artificial Intelligence Methods, *SAE International Journal of Advances and Current Practices in Mobility* 04 (4) (2022) 1072–1081. doi:10.4271/2022-01-0040.
- [7] Z. Wang, W. Yan, T. Oates, Time Series Classification from Scratch with Deep Neural Networks: A Strong Baseline, arXiv:1611.06455 [cs] (Dec. 2016). doi:10.48550/arXiv.1611.06455.
- [8] J. Mailhe, J. Linares, J. Sprauel, P. Bourdet, Geometrical checking by virtual gauge, including measurement uncertainties, *CIRP Annals* 57 (1) (2008) 513–516. doi:10.1016/j.cirp.2008.03.112.
- [9] J. G. Choi, D. C. Kim, M. Chung, S. Lim, H. W. Park, Multimodal 1D CNN for delamination prediction in CFRP drilling process with industrial robots, *Computers & Industrial Engineering* 190 (2024) 110074. doi:10.1016/j.cie.2024.110074.
- [10] W. Liu, J. Cui, Y. Lu, P. Yin, L. Han, Y. Jiang, Y. Zhang, Online prediction of composite material drilling quality based on multi-sensor fusion, *Journal of Intelligent Manufacturing* (Oct. 2024). doi:10.1007/s10845-024-02503-2.
- [11] G. Chen, Z. Chen, W. Peng, W. Wang, H. Zhao, Surface roughness classification of drilling for CNC machine tools based on RepViT and dual-channel STFT-GAF feature fusion, *Measurement Science and Technology* 36 (8) (2025) 085102. doi:10.1088/1361-6501/adeffe.
- [12] Z. Wang, T. Oates, Imaging Time-Series to Improve Classification and Imputation, arXiv:1506.00327 [cs] (Jun. 2015). doi:10.48550/arXiv.1506.00327.
- [13] A. Wang, H. Chen, Z. Lin, J. Han, G. Ding, RepViT: Revisiting Mobile CNN From ViT Perspective, arXiv:2307.09283 [cs] (Mar. 2024). doi:10.48550/arXiv.2307.09283.
- [14] J. G. Choi, D. C. Kim, M. Chung, G. Kim, H. W. Park, S. Lim, Accurate synthesis of sensor-to-machined-surface image generation in carbon fiber-reinforced plastic drilling, *Expert Systems with Applications* 255 (2024) 124656. doi:10.1016/j.eswa.2024.124656.
- [15] H. Ismail Fawaz, G. Forestier, J. Weber, L. Idoumghar, P.-A. Muller, Deep learning for time series classification: a review, *Data Mining and Knowledge Discovery* 33 (4) (2019) 917–963. doi:10.1007/s10618-019-00619-1.
- [16] A. Paszke et al., PyTorch: An Imperative Style, High-Performance Deep Learning Library, arXiv:1912.01703 [cs] (Dec. 2019). doi:10.48550/arXiv.1912.01703.
- [17] D. P. Kingma, J. Ba, Adam: A Method for Stochastic Optimization, arXiv:1412.6980 [cs] (Jan. 2017). doi:10.48550/arXiv.1412.6980.