Production Engineering
https://doi.org/10.1007/s11740-024-01266-2

PROCESSES

# Classification of the machine state in turning processes by using the acoustic emission

Daniel Diaz Ocampo[1] · Daniel Aubart[2] · Germán González[2] · Frederik Zanger[2] · Michael Heizmann[1]

Received: 31 August 2023 / Accepted: 15 January 2024

## Abstract

Processing digital information stands as a crucial foundation of Industry 4.0, facilitating a spectrum of activities from monitoring processes to their understanding and optimization. The application of data processing techniques, including feature extraction and classification, coupled with the identification of the most suitable features for specific purposes, continues to pose a significant challenge in the manufacturing sector. This research investigates the suitability of classification methods for machine and tool state classification by employing acoustic emission (AE) sensors during the dry turning of Ti6Al4V. Features such as quantiles, Fourier coefficients, and mel-frequency cepstral coefficients are extracted from the AE signals to facilitate classification. From this features the 20 best are selected for the classification to reduce the dimension of the feature space and redundancy. Algorithms including decision tree, k-nearest-neighbors (KNN), and quadratic discriminant analysis (QDA) are tested for the classification of machine states. Of these, QDA exhibits the highest accuracy at 98.6 %. Nonetheless, an examination of the confusion matrix reveals that certain classes, influenced by imbalanced training data, exhibit a lower prediction accuracy. In summary, the study affirms the potential of AE sensors for machine state recognition and tool condition monitoring. Although QDA emerges as the most acurate classifier, there remains an avenue for refinement, particularly in training data optimization and decision-making processes, to augment accuracy.

**Keywords** Classification · Machine state · Acoustic emission · Turning

## 1 Introduction

Advanced condition monitoring techniques are employed to supervise the machining process regarding the machine tool, workpiece, and cutting tools, with a primary focus on error detection [1] and tool wear detection [2]. Established analytical methods, including support vector machines, k-nearest-neighbors, decision trees, and artificial neural networks, have been predominantly used for fault detection of machining processes [3]. Techniques that are independent of specific models for configuring the hyper-parameters of monitoring systems through the utilization of look-up tables are commonly used in the industry [4]. Conventionally, both time and frequency domain features of sensor signals have been crucial for machine fault detection, including acoustic emissions and overarching machine parameters [5].

Methods like these focus on monitoring the process while the tool is engaged in the workpiece, so the use of such a system should be avoided if the tool is broken or the machine is in the wrong state. To achieve a more precise use of the process monitoring the knowledge of the current machine state is necessary [6]. Additionally, the machine state is an important factor in many fields like smart manufacturing or industry 4.0 [7].

The foundation of this work is also a process monitoring system, using acoustic emission sensors for observation of residual stress states in a longitudinal turning process of titanium alloy [8]. In the machining of titanium alloys, the physical mechanism of chip formation must be considered for accurate interpretation of data measured during process monitoring. Thermal softening leads to adiabatic shear bands, resulting in saw-tooth-shaped chip formation [9], a phenomenon known as chip segmentation. This specific

✉ Daniel Diaz Ocampo
daniel.diaz-ocampo@kit.edu

[1] Institute of Industrial Information Technology (IIIT), Karlsruhe Institute of Technology (KIT), Hertzstr. 16, Karlsruhe 76187, Germany

[2] wbk Institute of Production Science, Karlsruhe Institute of Technology (KIT), Kaiserstr. 12, Karlsruhe 76131, Germany

Published online: 29 February 2024

morphological structure of chips significantly affects the thermomechanical dynamics at the interface between the workpiece and cutting tool. Consequently, this influences the material removal rates and the overall dynamic behavior of the machining system [10]. The formation of serrated chips generates non-cyclical mechanical waves, referred to as chip segmentation frequency, detectable by acoustic emission sensors [11]. These frequencies can be correlated with other process information, such as surface integrity and can be measured by acoustic emission sensors [12].

For the evaluation of for example the chip segmentation frequency, it is extremely important to know in which state the machine is, especially when the tool is cutting. It therefor makes sense to use the already available acoustic emission sensor signals to estimate the current machine state, instead of installing additional hardware. The combination of a process monitoring system with a machine state classification system using the same sensors provides a good opportunity to retrofit legacy machines, which is an important task for smart manufacturing [13].

In this manuscript, established classification methods are applied and tested to estimate the current machine state in a longitudinal turning process of Ti6Al4V using acoustic emission sensors. In particular, the states "Off", "Idle", "Contact", "Cut", "Break", "Cut-Break" and "Reverb" have to be determined. Initially, the features extractable from AE signals, which will be utilized for monitoring the process, are introduced. Since the chip segmentation leads to a characteristic form of the structure-borne sensor signals, it is necessary to investigate special features for an optimal result. Subsequently a feasible way of feature selection is demonstrated to get the most efficient features. Then three common methods: Decision Tree, K-Nearest Neighbor, and Quadratic Discriminant Analysis to classificate the machine state based on the extracted features are described. Finally, the accuracy of these classification techniques in assessing machining states is evaluated. Since the method contains an automated feature selection and in addition an evaluation of the classification algorithm, it can be applied with additional training to other process parameters, materials or cutting processes.

## 2 Experimental Setup

Dry turning experiments on Ti6Al4V were executed utilizing an Index V100 vertical turning machine. The cutting tool employed was an uncoated carbide insert, type CCMW120404 with a cutting edge radius ($r_\beta$) of 50 µm. The geometrical specifications of the tool holder set the tool cutting edge angle ($\kappa_r$) at 95°, rake angle ($\gamma$) at 0°, and a clearance angle ($\alpha$) of 7°. During these machining tests, process control variables such as cutting speed ($v_c$), feed rate ($f$) and cutting depth ($a_p$) have been investigated as summarized in Table 1. The different sets of process control variables, when combined with inherent disparities in measurement configurations, induce variations in the acquired signals, spanning amplitude, waveform morphology, spectral diversity, among other attributes.

For discerning chip segmentation frequencies, a triad of piezoelectric acoustic emission (AE) sensors, the VS12-E model by Vallen Systeme, were mounted onto the tool holder aligned with the cutting trajectory, as visualized in figure 1. The raw AE sensor output underwent preliminary processing via an AE preamplifier, subsequently captured by a NI PXI station with a sampling frequency of 1 MHz at 16 bit resolution. Ambient acoustic detection was facilitated using PCB-378C01 microphones from PCB SYNOTECH GmbH, operational within a 6.3–126 kHz frequency spectrum. The comprehensive experimental procedure, spanning from real-time data collection to subsequent data processing, was optimized to fall within a 10 to 100 ms time frame. Throughout the machining phase, while the tool's cutting edge remained stationary, the spatial coordinates (x, y, z) of the workpiece underwent dynamic modifications.

## 3 Features

The concept of this work involves extracting multiple features from the acoustic emission signals, taken to classify the machine state, rather than using the raw signals. Here, it will be examined how this approach works in a longitudinal cutting process of titanium alloy for classification of machine states. Some of the features to estimate segmentation frequency were developed specifically for this process. Consequently, a large number of features were evaluated, and only those intended for use are described here.

### 3.1 Preprocessing and data structure

It is not feasible to calculate the features from an entire signal, as this approach would not be suitable for an online classifier. Instead, signals from different sensors are segmented into shorter sections, called "chunks" in this work, which are not overlapping. Investigations have shown that a chunk size of 8.192 , equivalent to a time span of 8.192 ms, yields optimal results with the following algorithms. The task is now to classify the machine state for each individual chunk, for which purpose features are calculated from each chunk, forming the basis for subsequent classification.

**Table 1** Process control variables

| $v_c$ (m/min) | $f$ (mm/rev) | $a_p$ (mm) |
|---|---|---|
| 40–120 | 0.08–0.20 | 0.25–1 |

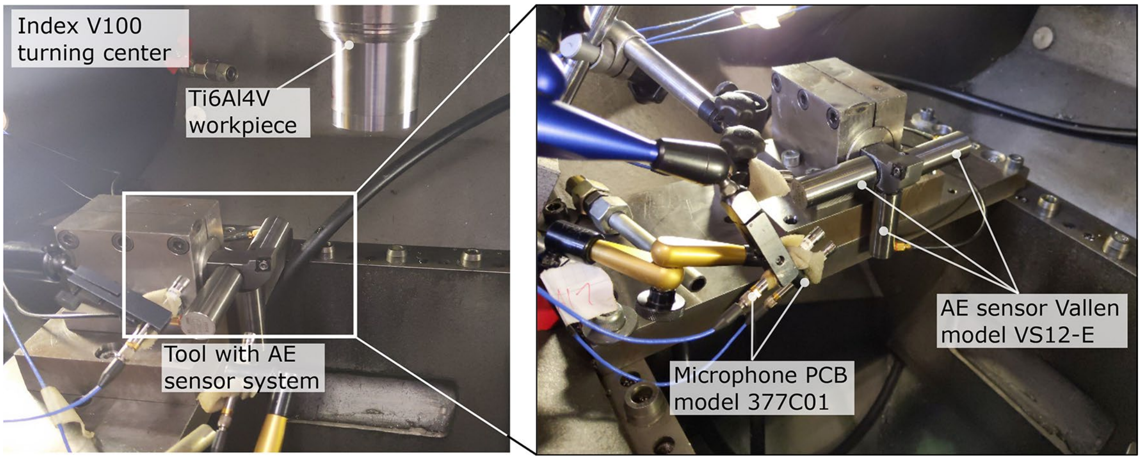

**Fig. 1** Set-Up AE sensors

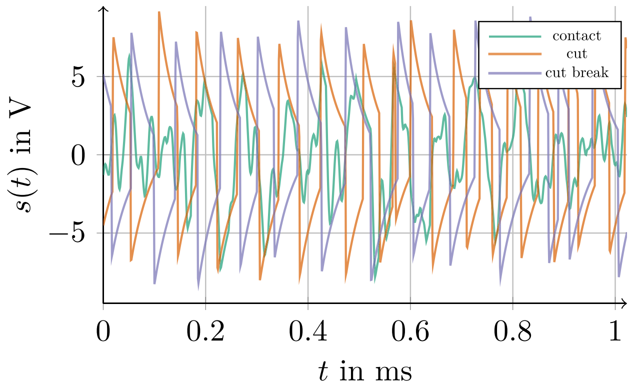

**Fig. 2** Acoustic emission example signal of the structure-borne sensor

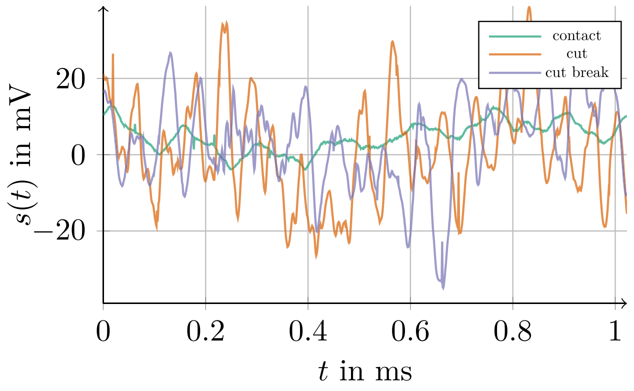

**Fig. 3** Acoustic emission example signal of the microphone

In Figs. 2 and 3 examples of the structure-borne signals respectively microphone signals are shown. It can be seen, that the structure-borne signals have a characteristic sawtooth form during cutting. In the following sections, several features are presented that are specifically suitable for this characteristic form. For both signal types it is not easy to notice a difference between the cut and the cut with broken tool. Also, there are signals where contact and cut look almost the same.

To ensure a robust classification considering sensor positioning, particularly in scenarios with multiple sensors of the same type, the data have been reorganized. Consequently, the signals have been divided into data from microphones and structure-borne sensors along the y-direction. From each chunk, all possible combinations of one structure-borne signal and one microphone signal are taken as training and test samples, respectively.

## 3.2 Quantile

An $X\%$-quantile is the threshold at which a certain percentage of the data set have a smaller or equal value. This helps to analyze the data distribution and offers insights into its central tendency and spread. To calculate the quantiles the absolute values of the signals are taken. Subsequently, quantiles ranging from 10 % to 90 %, with a step size of 10 %, are selected as features.

### 3.2.1 Spectral energy

The information of signals is often located across various spectral ranges. Therefore, filtering different frequency ranges of the raw signal is an obvious feature. To achieve this, three different infinite impulse response (IIR) filters are employed: a lowpass, a bandpass, and a highpass filter. Each filter employs four IIR coefficients and follows a Butterworth filter design, as described in [14]. The selection of cutoff frequencies ensures immediate adjacency of neighboring filters. Specifically, cutoff frequencies of 16.66 kHz and

33.33 kHz are utilized. To create a singular feature, the mean value and standard deviation of the absolute values derived from the resulting filtered signals are computed.

## 3.3 Fourier Transform

To convey fundamental properties of the spectrogram to the classifier, the Fast-Fourier-Transform is calculated from the signals as described in [15]. To get less but more robust features, with respect to signal noise, the result is divided into 20 equally sized frequency bins, and the corresponding absolute values are summed for each bin. These 20 values are normalized such that their sum equals one, representing the percentage contribution of each amplitude to the total sum. This approach allows the spectrogram to be represented in a computationally efficient manner and compensates for minor frequency variations through the aggregation into frequency bins. These values are particularly relevant for identifying whether vibrations of specific frequencies are present or not. This is especially applicable in fracture detection, as a noticeable frequency shift occurs after a fracture in the cutting edge.

## 3.4 Spectral flatness

To determine how much a signal is overlaid with white noise, the spectral flatness can be calculated as described in [16]. This is useful for detecting the states "Off" and "Idle". To calculate the value, the geometric mean of the amplitudes is divided by the arithmetic mean. If all amplitudes have very similar values, the geometric mean approaches the arithmetic mean, and the spectral flatness becomes close to one. The signal consists of pure noise. If the amplitude values differ significantly, the value of spectral flatness approaches zero.

## 3.5 Spectral roll-off

The term "Spectral Roll-off" encompasses the roll-off frequencies [17]. These frequencies indicate how much energy lies within the range below them. With an 85 %-roll-off frequency, for instance, 85 % of the signal's energy is located below that frequency. This can be utilized to differentiate between the two states of "Off" and "Idle" from the states of "Contact" and "Cut", as larger amplitudes occur at higher frequencies in the latter states.

To compute these frequencies, the energy of the spectrum is calculated for each range. In a second step, the energy of the frequencies below each frequency range is summed. Finally, the summed energies are divided by the total energy of the signal to determine the percentage.

For this work, the roll-off frequencies were calculated in 10 % increments from 50 % up to 90 %.

### 3.5.1 Mel-frequency cepstral coefficients

Mel-frequency cepstral coefficients (MFCCs) are used to condense a spectrum into a few key parameters. To achieve this, a spectrum is computed on the mel scale, and then transformed using a cosine transformation into a set of coefficients as described in [18]. In the first step, the spectrum is calculated by applying a discrete Fourier transformation to the chunks filtered with a Hanning window. The result of this transformation can also be interpreted as Short-Time-Fourier-Transform of the signal and yields a list of amplitudes $X_k$ with corresponding frequencies $\omega_k$. Subsequently the energy in each channel is calculated, where one channel is realized by triangular filters in the frequency domain. These filters are overlapping and their distance and bandwith increase with higher frequencies. The split into these channels can be interpreted as mapping the frequencies on a mel scale. The number of computed features per chunk depends on the number of channels chosen for this step. Now the logarithm to base 10 is calculated for each energy per channel. In the final step, a discrete cosine transformation (DCT) is performed, using a Type II DCT in practice. In addition to the original signal, MFCCs are also computed for the first and second derivatives.

### 3.5.2 Derivatives

During the cutting process, the signal from the structure-borne sensors exhibits a characteristic sawtooth pattern. Due to this sawtooth characteristic, the structure-borne signals exhibit sharp edges. These sharp edges potentially carry substantial process-related information. To capture this information, the absolute values of the discretized derivative using the difference quotient must be taken into account. For generating singular features per chunk, both the mean and the standard deviation are computed. Moreover, from the second derivative also the same features can be computed.

## 3.6 Sawtooth recognition

In Fig. 2 it can be seen, that the sawtooth mainly occurs during the cutting in structure-borne signals, but only rarely in contact, its detection proves to be a valuable feature. This sawtooth pattern is easily distinguished by its pronounced peaks, which can be effectively identified through its second derivative as described before. A chunk is classified as a sawtooth when the 98-percentile of the signal's second derivative exceeds a predetermined threshold. In practical testing, a threshold value of 0.1 has proven appropriate for this application, with the chunk undergoing normalization to a range between $-1$ and 1 before differentiation.

### 3.7 FIR filter

Investigations have shown, that in the structure-borne signals as can be seen in Fig. 2 the positions of the peaks carry substantial information, whereas the subsequent fading does not. To filter the fading a simple finite impulse response filter (FIR) can be applied. A FIR filter with two coefficients, namely −1 and 0.969, is sufficient to get a resulting signal with pronounced peaks at the signal's step positions. Subsequently, both the mean and standard deviation of the absolute values are computed for a single chunk.

### 3.8 Selection

Since 105 different features have been computed at this point it is not recommended to use all of them. Features can be heavily overlaid with noise, so their use in a classifier with distance calculation introduces noise into the classification process. Also features often contain the same information and are thus redundant. Not all features have been explained in this work, just those that are used.

The analysis of variances (ANOVA) encompasses methods for investigating the dispersion of variables in datasets and can help to choose the best features as described in [19]. The goal is to identify variables that provide high information content about class membership. Using the variances determined between classes $\sigma_{\mathrm{b}}^2$ and within classes $\sigma_{\mathrm{w}}^2$, the ability of variables to partition the dataset into classes can be evaluated. For a number of $M$ classes $C_1, \dots, C_M$, with $N_m$ training samples per class the F-score is calculated by

$$F = \frac{\sigma_{\mathrm{b}}^2}{\sigma_{\mathrm{w}}^2} = \frac{\sum\limits_{m=1}^{M} N_m (\bar{x}_m - \bar{x})^2 / (M-1)}{\sum\limits_{m=1}^{M} \sum\limits_{n=1}^{N_m} (x_{mn} - \bar{x}_m)^2 / \sum\limits_{m=1}^{M} (N_m - 1)}, \tag{1}$$

where $\bar{x}$ is the overall mean and $\bar{x}_j$ is the mean of a single class. $F$ indicates the proportion of dispersion within the classes to the total variance. The objective is to find a set of variables with minimal dispersion within the classes while maximizing the separation of class means. The $k$ variables with the highest F-scores are taken for the classification task.

## 4 Classification

The main purpose of the classification approach is to decide in which state the machine is, based on the calculated features of the chunks. For this work, seven different states have been considered. If the machine is turned off, there could be some noise from, for instance other machines in the same room, and if it is on but nothing is moving, there is noise from the hydraulic system of the machine. For the process, both cases are defined as the state "Off". If the workpiece is turning but nothing else moves, or if the workpiece is moving but there is no contact between the tool and workpiece, the state is defined as "Idle". In some cases, the workpiece has a step whereat the rear part has to be cut. Then, in the front part of the workpiece, the tool and the workpiece are in contact without cutting. So this state is defined as "Contact". The main state and the aim of this work is the state "Cut", which is the obvious state. Since the process is tested in a wide field of parameters, tool breaks are inevitable. But the moment of a tool break is quite short, which leads to an imbalanced dataset. There are only a few chunks where the tool breaks occur, whereas there are thousands of chunks for all the other states, which is a bad condition for the training of the classifiers. So in the case of a tool break, there are two states defined. At first, the "Break", and second, the cut with a tool break called "Cut-Break". For the second state, a big amount of examples exists, so the algorithms can be trained to recognize this state. After the cut, there are short moments where the machine does not cut, but the sensors measure acoustic emissions. This happens because of reflections of the airborne and structure-borne signals. These short states are defined as "Reverb".

### 4.1 Decision tree

A decision tree classifier aims to partition the training dataset into different classes using a decision tree structure, thereby learning classification rules. A decision tree consists of a root node, internal nodes, and multiple leafs as described in [20]. Each classification begins at the root node. At each node, a binary decision is made based on one single feature, if it is higher or lower than a threshold. The threshold and the chosen feature are constant and learned in the training, while the value of the feature varies for every chunk. Depending on the decision of the node, the next decision is made based on the feature and the threshold of the next node. This procedure is repeated until the end of the nodes, a so called leaf, that is assigned to a class is reached. Normally a maximum depth is given, but not all branches of the tree have to have this maximum number of decisions.

A decision tree is extremely simple and fast in inference, the difficulty is to find out which are the best features and their associated thresholds on which nodes. During the creation of the decision tree, the training dataset is divided into subsets at each node. Therefore the Gini index

$$G = \sum_{m=1}^{M} p_{jm}(1 - p_{jm}) \tag{2}$$

is used. For a given feature and a threshold the probability $p_{jm}$ is defined as the amount of training observations with the threshold with index $j$ of class $C_m$. The Gini index is small if all $p_{jm}$ are near 0 or 1, which is equivalent with a bad classification. In an iterative process, the feature and threshold with the highest Gini index at the current node is selected. To prevent overfitting, an early stopping criterion should be applied to the decision tree. It makes sense to set a minimum Gini index, if the result is below this, a node is defined as a leaf. Additionally often a maximum depth of the decision tree is defined.

### 4.2 K-nearest neighbor

The k-nearest-neighbor (KNN) classifier is one of the simplest and fastest classification algorithms to implement. It operates by classifying a sample of unknown class based on the majority consensus of its $k$ nearest neighbors, as described in [21].

Fundamental parameters that need to be defined during the classifier's design include the value of $k$, representing the number of nearest neighbors, and the formula for calculating the distance. Depending on the value of $k$, the class boundaries of the KNN classifier can be more or less smoothed, however with lower values of $k$ leading to a risk of overfitting. For distance calculation, it is important to normalize the features' variances beforehand to ensure consistent feature weighting. Various approaches for distance calculation can also be employed as described in [22]. In most cases, the Euclidean distance

$$d(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{n=1}^{N_f} (x_n - y_n)^2} \tag{3}$$

between two vectors $\mathbf{x}$ and $\mathbf{y}$ that represent the $N_f$ different features of two samples, is a sufficient choice and should only be deviated from in specific scenarios.

For classification using the KNN algorithm, the distance between the target sample and all stored samples in the classifier must be computed. This computation can be computationally intensive and memory-demanding in the implementation if too many training samples exist. Hence, efficient algorithms can be used to fasten the computation as for example the ball tree [23] or the k-d tree [24].

It is important to ensure that the training dataset is representative for the intended application, as the classification process relies on comparing with historical values rather than learning a strict "rule".

### 4.3 Quadratic discriminant analysis

Quadratic discriminant analysis (QDA) is a classical statistical method used in pattern recognition and classification tasks as described in [25]. It is closely related to linear discriminant analysis (LDA) but differs in its underlying assumptions and modeling approach. Quadratic discriminant analysis is a powerful technique for classification tasks, particularly when dealing with non-linear decision boundaries and varying covariance structures among classes. QDA, like LDA, aims to find a discriminant function that maximizes the separation between different classes in a given dataset.

For a sample $\mathbf{x}$, QDA models the conditional probability belonging to class $C_m$ as a multivariate Gaussian distribution, with its own covariance matrix $\boldsymbol{\Sigma}_m$ and mean vector $\boldsymbol{\mu}_m$ per class. Based on this model, it can be shown that the most likely class is the one that maximizes the term

$$\begin{aligned}\delta_m(\mathbf{x}) = & -\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_m)^T \boldsymbol{\Sigma}_m (\mathbf{x} - \boldsymbol{\mu}_m) \\ & - \frac{1}{2}\log\left|\boldsymbol{\Sigma}_m\right| + \log p_m,\end{aligned} \tag{4}$$

where $p_m$ is the prior probability that a sample $\mathbf{x}$ belongs to the $m$-th class. In the training of the QDA, the samples are used to estimate the parameters of Eq. (4).

$$\hat{p}_m = \frac{N_m}{\sum_{m=1}^{M} N_m} \tag{5}$$

$$\hat{\boldsymbol{\mu}}_m = \frac{1}{N_m} \sum_{n=1}^{N_m} \mathbf{x}_n \tag{6}$$

$$\hat{\boldsymbol{\Sigma}}_m = \frac{1}{N_m} \sum_{n=1}^{N_m} (\mathbf{x}_n - \hat{\boldsymbol{\mu}}_m)(\mathbf{x}_n - \hat{\boldsymbol{\mu}}_m)^T \tag{7}$$

Due to the model, QDA can provide accurate results when the class distributions have different covariance structures. However, QDA has its limitations as well. One major drawback is that it requires estimating a separate covariance matrix for each class, which becomes computationally expensive as the number of features increases. This can lead to overfitting when dealing with high-dimensional data. Additionally, QDA assumes that the class distributions are Gaussian, which might not hold true for all features.

## 5 Results

In the following section, the results and a comparison between the three methods are presented, as well as their advantages and disadvantages.

### 5.1 Selected features

At first, it is shown which features are selected by ANOVA, which is described in Sect. 3.8. Out of the 105 computed features, 20 have been selected. Most features are based on the structure-borne signals, which is as expected, since they are less disturbed by other effects and their volume is a good indicator for the contact between tool and workpiece. Also, as expected, many features are taken from specific frequency bands. From the sensor signals, the following features are selected:

- Structure-borne: Quantile (10%)
- Structure-borne: Derivative 2 (std)
- Structure-borne: FIR (std)
- Structure-borne: Bandpass (mean, std)
- Structure-borne: Highpass (mean, std)
- Structure-borne: Fourier bins (6–11)
- Structure-borne: Spectral roll-off (80 %, 90 %)
- Structure-borne: Spectral flatness
- Structure-borne: Sawtooth
- Microphone: MFCC (0–2)

As it can be seen from the list of selected features, the selection is not optimal. The approach of ANOVA provides the most expressive features, but there are several features that are quite related. Many features like Fourier bins or bandpass observe the same frequency bands in a very similar way. This Problem can be solved in future by using for example MANOVA [26].

### 5.2 Classification performance

Several methods have been tested for this classification task, but only the three best have been introduced in Sect. 4. Others that have been tested, but have been worse or did not bring better results despite extra effort, are naive Bayes classifier, multilayer perceptron, support vector machine, and random forest.

To evaluate the performance of the classifier, two metrics have been chosen. First of all the correct classification rate

$$R_{\text{true}} = \frac{N_{\text{true}}}{N_{\text{test}}} \cdot 100\,\% \tag{8}$$

has been calculated as a crude evaluation metric, where $N_{\text{true}}$ is the number of correct classifications and $N_{\text{test}}$ is the total number of samples in the test dataset. To get the rate in percent, the fraction is multiplicated with 100 %.

The introduced methods have been tested with different parameters and the most relevant correct classification rates are

- KNN ($k = 4$): 91.5 %
- KNN ($k = 15$): 92.2 %
- Decision tree (depth = 5): 93.8 %
- QDA: 98.6 %.

It can be seen that a decision tree and a KNN have a quite similar performance. This is consistent for different parameters of both methods. Additionally, the KNN with 15 neighbors performs not that much better than the KNN with just 4 neighbors so a simple KNN with $k = 4$ should be preferred. Random forests have also been tested ($R_{\text{true}} = 92.6\,\%$) but they perform not better than a simple decision tree, so the additional expense is not justified. The QDA performs much better than the other methods and has the best results.

For a detailed evaluation, the confusion matrix can be taken. In Table 2 the confusion matrix of the QDA is shown. The first line shows the amount of training examples $N_m$ and

**Table 2** Confusion matrix of the QDA classifier. The results are in % and the first row shows the number of samples $N_m$ in the training set per class

| Off<br>4851 | Idle<br>76,296 | Contact<br>390,804 | Cut<br>63,894 | Break<br>186 | Cut-Break<br>27,840 | Reverb<br>240 |
|---|---|---|---|---|---|---|
| 100.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| 0.0 | 99.2 | 0.8 | 0.0 | 0.0 | 0.0 | 0.0 |
| 0.0 | 0.1 | 99.9 | 0.0 | 0.0 | 0.0 | 0.0 |
| 0.0 | 0.0 | 0.0 | 98.5 | 0.0 | 1.5 | 0.0 |
| 0.0 | 0.0 | 0.0 | 75.0 | 0.0 | 25.0 | 0.0 |
| 0.0 | 0.0 | 0.1 | 81.6 | 0.0 | 18.3 | 0.0 |
| 0.0 | 0.0 | 100.0 | 0.0 | 0.0 | 0.0 | 0.0 |

every subsequent line shows the classes in the same order like the columns.

At first, it can be seen that the amount of training data is not balanced over the classes, which leads to unequal treatment of the classes by the classifier. The reason for this is that, for example, a tool break is just a short moment in a whole cut, so there are thousands of examples for "Cut" respectively "Cut-Break", while there is just one example for "Break". This is why all classifier have a bad performance for this class, which is often classified as "Cut" or "Cut-Break" and only the QDA recognizes some of the tool breaks. To recognize a tool break, the class "Cut-Break" performs way better. This class is also the source for the better performance of the QDA. The KNN (39.5 %) and the decision tree (42.7 %) are not able to recognize this class in the required quality. It would be a better approach for KNN and decision tree to split the problem to train a classifier just for tool breaks, especially since a tool break can be recognized in the microphone signals as a high peak.

A similar problem exists with the class "Reverb", which also has not many examples. None of the methods recognizes this class, but since this is no important class in practice, it has no bad effects. Hence, the samples of the class "Reverb" can be ignored.

The performance for the main classes is very good. To prove this, the error rates have been calculated again, without the classes "Break" and "Reverb". Additionally, the class "Cut-Break" has been merged together with the class "Cut". Without a new training of the methods the resulting classification rates are

- KNN ($k = 4$): 95.6 %
- KNN ($k = 15$): 98.3 %
- Decision tree (depth = 5): 99.6 %
- QDA: 98.9 %.

Especially the classes "Contact" and "Cut" can be separated really good and in this case the decision tree has the best overall performance. The remaining errors can be improved by using multiple decisions of the classifier to implement a filtering over time.

## 6 Conclusion

In conclusion, this study investigates the suitability of methods for the machine state classification using acoustic emission sensor data on the example of longitudinal turning of titanium alloy. As discrete machine states, among others "Contact" or "Cut" and in addition tool states like "Break" or "Cut-Break" were defined. Experiments on a turning machine Index V100 were conducted, extracting various features from structure-borne and microphone signals. Decision trees, k-nearest-neighbors, and quadratic discriminant analysis were employed for classification, with QDA demonstrating superior performance due to its ability to model diverse covariance structures. Challenges arose with limited examples in certain classes. While QDA exhibited potential, future work could involve rebalancing techniques, ensemble methods, and temporal filtering to enhance results. The goal was to extend an acoustic emission system for process monitoring to classify the machine states, without implementing additional hardware. If the importance of the classes is taken into account, the decision tree provides the best results of 99.6 % so the goal is achieved.

**Acknowledgements** The authors gratefully acknowledge the financial support of the German Research Foundation, Deutsche Forschungsgemeinschaft (DFG) within the research priority program SPP2086 "Surface Conditioning in machining" SPP 2086 (ZA 785/3-2 and HE 6212/1-2).

**Funding** Open Access funding enabled and organized by Projekt DEAL. .

## Declarations

**Conflict of interest** The authors declare that they have no conflict of interest.

## References

1. Huang H, Zhao HHL, Guo S (2020) Machine fault detection for. IEEE Commun Mag 58(1):40–46
2. Mohamed A, Hassan M, M'Saoubi R, Attia H Tool condition monitoring for high-performance machining systems—a review **22**(6):2206. https://doi.org/10.3390/s22062206
3. Saufi SR, Ahmad ZAB, Leong MS, Lim MH (2019) Challenges and opportunities of deep learning models for machinery fault detection and diagnosis: A review. IEEE Access 7(7):122644–122662
4. Denkena B, Klemme H, Becker J, Blech H (2022) User-specific parameterization of process monitoring systems 16(6):735–742. https://doi.org/10.1007/s11740-022-01130-1
5. Widodo A, Yang B-S (2007) Support vector machine in machine condition monitoring and fault diagnosis. Mech Syst Signal Process 21(6):2560–2574
6. Kuppers F, Albers J, Haselhoff A Random forest on an embedded device for real-time machine state classification. In: 2019 27th European Signal Processing Conference (EUSIPCO). IEEE. https://doi.org/10.23919/eusipco.2019.8902993

7. Kummer A, Ruppert T, Medvegy T, Abonyi J Machine learning-based software sensors for machine state monitoring - the role of smote-based data augmentation 16:100778. https://doi.org/10.1016/j.rineng.2022.100778
8. Pachnek F, González G, Ocampo DD, Heizmann M, Zanger F Analysis of chip segmentation frequencies in turning ti-6al-4v for the prediction of residual stresses 108:188–193. https://doi.org/10.1016/j.procir.2022.03.033
9. Komanduri R (1982) Some clarifications on the mechanics of chip formation when machining titanium alloys. Wear 76(1):15–34
10. Melkote SN, Liu R, Fernandez-Zelaia P, Marusich T (2015) A physically based constitutive model for simulation of segmented chip formation in orthogonal cutting of commercially pure titanium. CIRP Ann 64(1):65–68. https://doi.org/10.1016/j.cirp.2015.04.060
11. Denkena B, Bergmann B, Schaper F Investigation of chip formation of Ti–6Al–4V in oxygen-free atmosphere 124(10):3601–3613. https://doi.org/10.1007/s00170-022-10655-9
12. Schwär D, González G, Segebade E, Zanger F, Heizmann M (2020) Evaluation of the acoustic emission caused by the chip segmentation frequency during machining of titanium alloy. tm - Technisches Messen 87(11):714–720. https://doi.org/10.1515/teme-2020-0056
13. Jaspert D, Ebel M, Eckhardt A, Poeppelbuss J Smart retrofitting in manufacturing: a systematic review. 312:127555. https://doi.org/10.1016/j.jclepro.2021.127555
14. Selesnick IW, Burrus CS (1998) Generalized digital butterworth filter design. IEEE Trans Signal Process 46(6):1688–1694
15. Brigham EO, Morrow R (1967) The fast fourier transform. IEEE Spectr 4(12):63–70
16. Dubnov S (2004) Generalization of spectral flatness measure for non-gaussian linear processes. IEEE Signal Process Lett 11(8):698–701
17. Kos M, Kačič Z, Vlaj D (2013) Acoustic classification and segmentation using modified spectral roll-off and variance-based features. Digit Signal Process 23(2):659–674
18. Vergin R, O'Shaughnessy D, Farhat A (1999) Generalized mel frequency cepstral coefficients for large-vocabulary speaker-independent continuous-speech recognition. IEEE Trans Speech Audio Process 7(5):525–532
19. St L, Wold S et al (1989) Analysis of variance (anova). Chemom Intell Lab Syst 6(4):259–272
20. James G, Witten D, Hastie T, Tibshirani R (2021). An introduction to statistical learning Springer. https://doi.org/10.1007/978-1-0716-1418-1
21. Cover T, Hart P (1967) Nearest neighbor pattern classification 13(1):21–27. https://doi.org/10.1109/tit.1967.1053964
22. Chomboon K, Chujai P, Teerarassammee P, Kerdprasop K, Kerdprasop N (2015) An empirical study of distance metrics for k-nearest neighbor algorithm. In: The Proceedings of the 2nd International Conference on Industrial Application Engineering. The Institute of Industrial Applications Engineers. https://doi.org/10.12792/iciae2015.051
23. Liu T, Moore AW, Gray A (2006) New algorithms for efficient high-dimensional nonparametric classification. The MIT Press. https://doi.org/10.7551/mitpress/4908.003.0008
24. Sproull RF (1991) Refinements to nearest-neighbor searching ink-dimensional trees 6(1–6):579–589. https://doi.org/10.1007/bf01759061
25. Ghojogh B, Crowley M (2019) Linear and quadratic discriminant analysis: tutorial. https://doi.org/10.48550/ARXIV.1906.02590
26. Bray JH, Maxwell SE (1985) Multivariate analysis of variance. Sage, Philippians