Nondestructive Acoustic Testing of Ceramic Capacitors Using One-Class Support Vector Machine With Automated Hyperparameter Selection

SAKU LEVIKARI\textsuperscript{1}, (Member, IEEE), TOMMI J. KÄRKKÄINEN\textsuperscript{1}, (Member, IEEE), CAROLINE ANDERSSON\textsuperscript{2}, (Member, IEEE), JUHA TAMMINEN\textsuperscript{3}, (Member, IEEE), MIKKO NYKYRI\textsuperscript{1}, (Member, IEEE), AND PERTTI SILVENTOINEN\textsuperscript{1}, (Member, IEEE)

\textsuperscript{1}LUT School of Engineering Systems, LUT University, 53850 Lappeenranta, Finland
\textsuperscript{2}ABB Schweiz AG, 5300 Turgi, Switzerland
\textsuperscript{3}ABB Oy Drives, 00380 Helsinki, Finland

Corresponding author: Saku Levikari (saku.levikari@lut.fi)

This work was supported by the Business Finland through the Project SolarX.

ABSTRACT The energy transition and electrification across many industries place increasingly more weight on the reliability of power electronics. A significant fraction of breakdowns in electronic devices result from capacitor failures. Multilayer ceramic capacitors, the most common capacitor type, are especially prone to mechanical damage, for instance, during the assembly of a printed circuit board. Such damage may dramatically shorten the life span of the component, eventually resulting in failure of the entire electronic device. Unfortunately, current electrical production line testing methods are often unable to reveal these types of damage. While recent studies have shown that acoustic measurements can provide information about the structural condition of a capacitor, reliable detection of damage from acoustic signals remains difficult. Although supervised machine learning classifiers have been proposed as a solution, they require a large training data set containing manually inspected damaged and intact capacitor samples. In this work, acoustic identification of damaged capacitors is demonstrated without a manually labeled data set. Accurate and robust classification is achieved by using a one-class support vector machine, a machine learning model trained solely on intact capacitors. Furthermore, a new algorithm for optimizing the classification performance of the model is presented. By the proposed approach, acoustic testing can be generalized to various capacitor sizes, making it a potential tool for production line testing.

INDEX TERMS Ceramic capacitors, acoustic emission, nondestructive testing, support vector machines.

I. INTRODUCTION

The ongoing energy transition has resulted in exponential growth in the market for power electronics, with applications such as inverters and drives gaining ground in renewable energy production and the electrification of transport [1]. This places more weight on the reliability of power electronics devices, as their abrupt failure will cause costly repairs, downtime, and in worst cases, even life-threatening situations. It has been estimated that 30\% of failures in electronics are caused by capacitors [2], the most widely used type of which is the multilayer ceramic capacitor (MLCC) with more than 10\textsuperscript{12} units produced yearly [3].

Cracks are the most prevalent failure mode in ceramic capacitors [4], The reason for this can be traced back to the structure of the MLCC, which consists of interleaved metal electrodes with ceramic dielectric (typically barium titanate, Ba\textsubscript{TiO\textsubscript{3}}) in between. The ceramic material gives the capacitors a high permittivity [5], but also makes them fragile. Thus, MLCCs can be damaged during the assembly of the printed circuit board (PCB), for example, as a result of thermal stresses during soldering or mechanical mishandling of the
circuit board. Common examples of stress-induced damage include cracks in the dielectric material and delamination between internal electrodes or between the capacitor body and its end terminations or solder joints [6]. Defects related to the manufacturing process of the component are also likely to increase the probability of damage during assembly [4]. Large case sizes (1812 and up) in particular have been shown to be prone to bending damage [7], and while the majority of the MLCC demand is focused on smaller case sizes found in consumer electronics and automotive industry [8], larger case sizes remain common in industrial applications and power electronics [9].

Detecting a faulty MLCC can be difficult, because the damage may not be visible on the outside of the component. If a crack occurs within the active area of the capacitor, for example cutting through a portion of the internal electrodes, the damage can be observed as a reduced capacitance. On the other hand, if a crack or delamination resides in the passive region of the MLCC or at the solder joints, the electrical operation of the component is likely to remain unaffected at first [10]. However, the damaged location may grow in size over time, leading to a reduced capacitance, open- or short-circuit [11], or deterioration of the dielectric material as moisture gets inside the capacitor [12], [13]. While the probability of failure for a single capacitor is very low, capacitor failures contribute to a significant fraction of faults in the field owing to the high number of components that are used [4]. A fast and reliable postassembly MLCC screening method would therefore be a highly valuable quality tool for quality assurance.

A. DETECTING DAMAGE IN MLCCs

Various methods for detecting damaged MLCCs have been proposed over the years, with a thorough survey on the topic available for instance in [4]. As pointed out in the survey, techniques commonly found in screening and quality assurance applications include electrical, optical, and non-contact ultrasound measurements. However, these methods are mainly applied to detecting manufacturing defects instead of damage caused later on in the production line. Moreover, the crack detection capability of these methods is limited, with electrical measurements often unable to detect cracks and optical methods limited to the surface of the component. C-scan ultrasonic microscopy has traditionally been capable of detecting only horizontally spanning defects in MLCCs, even though a newer technique capable of observing vertical damage has also been developed. However, this method is mainly used in R&D and pilot production [14] as it includes submerging the component in water, and it is not able to reveal damage in areas covered by end terminations [4].

For a more thorough inspection, a capacitor can be chemically etched [15] or mechanically microsectioned [16] in order to reveal damage in the ceramic body, but these methods are slow, laborious, and destructive to the component. Recently, X-ray imaging methods with sufficient accuracy for detecting cracks [16] and dielectric breakdown defects [17] have been developed. However, these methods require manual interpretation of the X-ray images. Moreover, delamination between the ceramic body and the solder joints cannot be seen by X-ray imaging, and cracks in the capacitor body are only visible from certain angles.

B. ACOUSTIC EMISSIONS IN MLCCs

MLCCs are known to generate acoustic emissions when subjected to alternating voltage. This phenomenon arises from the piezoelectric properties of the ceramic dielectric, making the capacitor body physically vibrate as the ceramic material deforms under an AC field [18]. To date, there are several studies about observing cracks in the ceramic body of an MLCC by changes in its resonant behavior [19]–[21]. These experiments are mainly based on ferroelectric transduction, where a DC-biased MLCC is excited using short radio frequency tone bursts, and the decay of the MLCC “ringing” after the burst is observed.

A more recent method introduced by the authors is based on electrically exciting the component to vibration, simultaneously measuring the vibrations using a point contact sensor [22]. However, while the acoustic emissions carry information about the structural condition of a capacitor, the signs of damage can be hard to detect. The amplitudes and frequencies of the mechanical resonances exhibited by MLCCs show variation even between intact components, and although mechanical damage often manifests as new resonant peaks, such peaks do not always appear [23]. Moreover, factors such as electromagnetic interference and variations in the mechanical contact between the sensor and the capacitor can cause artifacts that could be misinterpreted as signs of damage. Supervised machine learning algorithms were proposed in [23] as a solution on a small number of damaged MLCCs, yet the accuracy and applicability of the method remained inconclusive. A major disadvantage of such an approach is that a supervised machine learning model requires a large annotated data set on which the model is trained. This, in turn, requires acoustic measurements on a number capacitors, which are then labeled as intact or damaged. The labeling process itself has to be carried out manually for instance by microsectioning, which itself is a laborious and error-prone procedure.

In this study, the authors continue their prior work reported in [22] and [23] by showing that MLCC acoustic emission measurement can be used as a practical quality assurance testing method without the need for a laborious and error-prone labeling process. This is successfully demonstrated by identifying damaged MLCCs using the one-class support vector machine, a machine learning model trained only on pristine capacitor samples. Furthermore, a new algorithm for optimizing the hyperparameters of the model is introduced. By combining the proposed algorithm with proper feature extraction, the model can be made robust against error sources in the measurement setup and intrinsic variations between individual components.
II. METHODS AND RATIONALE

The task of acoustically detecting damaged MLCCs was approached as a machine learning problem, because it is infeasible to manually set the classification rules [23]. Machine learning models commonly employ a supervised learning strategy, in other words, the model is trained on a data set of annotated samples. In the case of detecting faulty MLCCs, a two-class classifier would be a natural solution, as the main interest is to know whether a capacitor is damaged (class 1) or not (class 0). To this end, the model would require a training data set consisting of MLCC acoustic signatures, each associated with a class label, 0 or 1. However, the process of constructing such a data set is work-intensive and prone to error, as a sufficiently large number (dozens or hundreds) of examples from both classes are needed for the training process. While the acoustic characterization process itself is relatively simple and fast, annotating the capacitor samples for instance by microsectioning or X-ray is time consuming. Visual interpretation of the results is also an error-prone process, which leads to training data with mislabeled examples.

This degrades the performance of the classifier and makes the evaluation of the model more difficult. Different failure modes are also likely to induce different characteristics in the acoustic response, and thus, dozens or hundreds of capacitors with cracks and delaminations are needed. Moreover, the size and internal structure of an MLCC affect both the acoustic characteristics and the most prevalent failure modes of the component. Thus, each MLCC size and capacitance should preferably have a data set of its own.

To overcome these limitations, a one-class support vector machine (OSVM), a machine learning model that learns from only a single class of examples, is employed. The model is used to determine whether a new acoustic signature is similar to known pristine signatures or not, thereby eliminating the need for annotated examples of damaged MLCCs. However, the performance of the model strongly depends on its hyperparameters, which are set before the learning process. Several methods for optimizing these parameters exist, but they typically require either beforehand knowledge of the classification problem, or counter examples, which in the case of MLCCs are costly to obtain. To facilitate fully automated hyperparameter optimization with no counter examples or a priori knowledge of the problem, a new algorithm is presented. The proposed approach can drastically reduce the amount of work on constructing a labeled data set of acoustic emissions, as only a few samples will be needed for testing the performance of the model. While the use of an OSVM in the context of screening MLCCs is novel, it has previously been used in other damage detection and condition monitoring applications, where data representing anomalous instances are difficult or expensive to collect [24]–[27].

The experimental section of this paper is organized as follows: first, an acoustic data set is composed of measurements performed by the authors in a prior study [23] on a set of intact and mechanically damaged MLCCs. For each capacitor, a set of numerical features are extracted from the raw acoustic data. Each MLCC is then attached with a label indicating whether the component is damaged or not. For automated detection of damaged MLCCs, an OSVM model is trained using data only from intact components. To achieve a high detection rate with few false alarms, an algorithm for optimizing the hyperparameters of the model is introduced and tested on commonly used benchmark data sets. Finally, the performance of the OSVM model is tested against damaged and undamaged capacitors.

A. ACOUSTIC EMISSION MEASUREMENTS ON MLCCs

The raw data used in this study originate from measurements performed by the authors in [22] and [23]. The measurements were conducted by subjecting soldered MLCCs to an excitation signal, simultaneously measuring the acoustic response of the capacitor using a piezoelectric point contact sensor placed on top of the component (see Fig. 1). An AC voltage chirp with pulsed waveform (duty cycle 80%, ±10 Vpeak-to-peak) was used as the excitation signal, with frequency linearly increasing from 100 Hz to 2 MHz over a duration of 100 ms. This study was performed on 2220-size MLCCs, as larger capacitors are more likely to be affected by damage related to soldering and assembly. The capacitors were from two manufacturers, all rated at 24 V and 22 µF, and soldered on the test PCBs in three different azimuthal orientations (0°, 45°, and 90°). The acoustic emission measurement setup is depicted in Fig. 2 along with other preprocessing and analysis steps, and the measurement equipment is listed in Table 1.

The capacitors were soldered on two identical custom-made test boards (PCB 1 and PCB 2) with 60 MLCCs each. Each capacitor was acoustically characterized, after which the capacitors on PCB 1 were mechanically damaged by subjecting the board to a controlled four-point bending setup with an average bending strain of 6000 µStr. The capacitors on PCB 1 were then recharacterized, cut from the PCB, and inspected for damage using cross-sectioning. Both cracks and delaminations were found during the inspection, with the length of the delaminations ranging from 200 µm to 500 µm.
FIGURE 2. Flowchart of the procedure for composing the acoustic emission (AE) data set (from top to bottom). Raw acoustic data, optical microscopy images, and X-ray images originate from prior work of the authors [22], [23]. The data set used in this study was composed by processing the raw acoustic data and inspecting the cross-sectional and X-ray images. A total of 180 MLCC samples from two test boards were included in the data set: 60 intact capacitors from PCBs 1 and 2 each, and 60 capacitors from PCB 1 after a controlled bending procedure. The extracted features and categorical labels were used as input and output variables for the OSVM model.

and the smallest cracks being only 100 µm in size. A selected set of the MLCCs were also inspected for cracks with X-ray imaging because their acoustic signatures appeared slightly anomalous. The cross-sectioning and X-ray imaging were performed in the same study as the acoustic measurements; more comprehensive descriptions of the methods are available in [22] and [23].

TABLE 1. Equipment used for Acoustic Emission measurements (refer to Fig. 2).

| Device                  | Make and model       |
|-------------------------|----------------------|
| Signal generator        | Agilent 33250A       |
| Point contact sensor    | KRN Services KRNB5-PC|
| Preamplifier            | KRN Services AMP-1BB-J|
| Oscilloscope            | Keysight InfiniiVision MSO-X 4104A|

FIGURE 3. Numerical features (see Table 2) were extracted from raw acoustic signatures (example shown in A). From the signal envelope (B), amplitudes and frequencies of the highest peaks below and above 700 kHz (the dashed red line) were used as features (marked with red circles). Additionally, the median amplitude and frequency of all resonance peaks in B were chosen as features. From the phase graph (shown in C), total phase shift and group delay ripple (zoomed-in section D) were used as features.

B. PREPROCESSING OF ACOUSTIC DATA

The raw acoustic waveforms (see Fig. 3A) were filtered using interval-dependent wavelet denoising. A second-order biorthogonal wavelet with eight vanishing moments was chosen for accurate signal approximation and smooth reconstructed waveform. After denoising, the waveforms were high-pass filtered in order to remove the high-amplitude burst below 150 kHz seen in Fig. 3A, likely related to PCB resonances [28], using a fourth-order Butterworth filter with a cutoff frequency of 150 kHz. After this, the harmonics caused by the pulsed waveform of the excitation signal were removed. To achieve this without losing the content of the acoustic signal at the excitation frequency, the signal was divided into 64 blocks with a 50% overlap. To facilitate an accurate overlap-add decomposition [29], each block was windowed using the von Hann function, and then low-pass filtered using a fourth-order Butterworth filter with a cutoff frequency of 1.3 times the excitation frequency at the endpoint of the block to remove the frequency content above the
first harmonic. Finally, the filtered signal \( u(t) \) was reconstructed from the filtered blocks using the overlap-add method. The aforementioned steps were performed using zero-phase forward-backward filtering to avoid distorting the phase of the signal.

After filtering, the envelope \( e(t) \) (see Fig. 3B) was calculated from the filtered signal \( u(t) \) using the Hilbert transform \( \mathcal{H} \) as

\[
e(t) = \text{DS} \left\{ \text{lpf} \left[ \sqrt{ (u(t))^2 + \text{Re} \{ \mathcal{H}(u(t)) \}^2 } \right] \right\},
\]

where DS and lpf denote downsampling and low-pass filtering, and \( \text{Re} \) denotes the real part of the Hilbert transform. The envelope can be seen as the acoustic amplitude response of the capacitor. In addition, the instantaneous phase response (see Fig. 3C) was calculated for each MLCC as

\[
\phi(t) = \text{DS} \left\{ \text{lpf} \left[ -\mathcal{H} \left( \frac{u(t)}{C(t)} \right) \right] \right\},
\]

where \( C \) is a linear sinusoidal chirp whose frequency sweep corresponds to the acoustic excitation signal. Finally, the resulting amplitude and phase responses were downsampled to 10,000 points each for feature extraction. All parameters related to signal processing were chosen empirically by examining acoustic data from both damaged and undamaged capacitors.

### C. FEATURE EXTRACTION

Feature engineering is a critical part of constructing a machine learning model, as its purpose is to reduce the dimensionality of the data and provide the model with only relevant information about the problem. To this end, a set of numerical features were extracted for each capacitor sample after preprocessing the raw acoustic waveforms.

While machine learning models using raw waveforms have been developed [30], such models are usually trained on large data sets with tens or hundreds of thousands of samples. Training a model with plain acoustic waveforms is unfeasible in this case, because the size of the data set is limited, and only a small fraction of the points in a single waveform are relevant. Moreover, the acoustic waveforms cannot be downsampled to under 500 points without losing the characteristics of the data, whereas most machine learning models perform poorly when the dimensionality of the data is higher than the number of samples [31].

Because the total number of capacitors was only 180, the number of features had to be limited to a number much smaller than the size of the data set. Furthermore, the features themselves had to be robust so that no external factor, such as EMI noise or variations in the sensor–capacitor contact, would bias the classification results. Based on these conditions, eight numerical features shown in Table 2 were extracted for each MLCC. The resonant peak amplitudes \( A_1 \) and \( A_2 \) along with their corresponding resonance frequencies \( f_1 \) and \( f_2 \) were chosen, as past studies [20, 21, 23] have suggested that these resonance peaks might indicate the presence of damage. However, especially \( A_1 \) and \( A_2 \) displayed notable systematic differences between two intact PCBs, probably related to external factors such as the level of ambient EMI, as the two boards were characterized on different occasions. Nevertheless, resonant peaks were assumed to be a very likely indicator of physical damage, and thus, the feature set was appended with the median amplitude \( m_A \) and frequency \( m_f \) of ten of the highest resonant peaks below 700 kHz. Furthermore, it was assumed that mechanical damage would cause distortions in the acoustic phase response of a capacitor body. Therefore, the total phase shift \( \Delta \phi \) was calculated for each capacitor. Finally, as an additional indicator of phase distortions, the mean group delay ripple \( GDR \) was calculated as the mean value of deviations from a linear frequency slope, as shown in Fig. 3D.

### TABLE 2. Features extracted from the acoustic responses.

| Feature | Description |
|---------|-------------|
| (1) \( A_1 \) | Amplitude of the highest peak below 700 kHz, typically found at approx. 500 kHz |
| (2) \( A_2 \) | Amplitude of the highest peak above 700 kHz |
| (3) \( f_1 \) | Frequency of \( A_1 \) |
| (4) \( f_2 \) | Frequency of \( A_2 \) |
| (5) \( \Delta \phi \) | Total phase shift calculated from (2) as \( \Delta \phi = \int f_1 |\phi(t)| dt \) |
| (6) \( m_A \) | Median amplitude of all resonance peaks in the acoustic spectrum |
| (7) \( m_f \) | Median frequency of all resonance peaks in the acoustic spectrum |
| (8) \( GDR \) | Mean group delay ripple. For each descending slope in the instantaneous phase curve, group delay ripple was calculated as the maximum deviation from the corresponding linear slope (see Fig. 3, bottom graph). The feature \( GDR \) was composed as the mean of all calculated group delay ripple values. |

### TABLE 3. Summary of the MLCC acoustic data set. The MLCCs on PCB 2 were used for training the OSVM model and hyperparameter optimization.

| PCB | Total samples | No damage \((y=0)\) | Damage \((y=1)\) |
|-----|---------------|-----------------|----------------|
| PCB 1 (before bending) | 60 | 60 | 0 |
| PCB 1 (after bending) | 60 | 23 | 37 |
| PCB 2 (no bending) | 60 | 60 | 0 |
| Total | 180 | 143 | 37 |

Finally, an acoustic data set (see Table 3) was composed by combining the extracted acoustic features with the microsectioning and X-ray results. Thus, the acoustic response of each MLCC is represented as a vector

\[
x_i = (A_1, A_2, f_1, f_2, \Delta \phi, m_A, m_f, GDR)
\]

and the data set \( X \) consists of pairs of feature vector \( x_i \) and class label \( y_i \)

\[
X = \{ (x_i, y_i) | x_i \in \mathbb{R}^8, y_i \in \{0, 1\} \}
\]
where \( i = 1 \ldots 180, y = 0 \) denotes nondamaged and \( y = 1 \) damaged MLCCs. All data in \( X \) were standardized to zero mean and unit variance according to the mean and variance of the data from PCB 2.

In order to avoid any external influences affecting the outcome of the tests, the model was trained on data from the intact PCB 2, and then tested on data from PCB 1 before and after bending. This approach ensures that the model is trained and tested on different individual components, which gives more realistic results on the performance of the model. The model was trained solely on data from pristine components (\( y = 0 \)), in which case the target labels can be considered accurate. However, the labels for the bent PCB 1 cannot be guaranteed to be fully accurate as the image inspection was done by eye, and small cracks and delaminations, in particular, can be difficult to spot. The acoustic data and target labels for PCB 2 (after bending) were also used as the basis of the analysis in a prior publication [23].

**D. ONE-CLASS SUPPORT VECTOR MACHINE**

As the motivation of this work was to distinguish damaged MLCCs based solely on intact examples, common supervised machine learning classifiers were unsuitable for the task. Instead, an outlier detection approach was employed using the one-class support vector machine (OSVM), which uses data from only a single class (intact capacitors) during the learning phase. Thus, the one-class SVM has been regarded as a semisupervised method [32], [33], even though the term semisupervised commonly refers to methods that use both labeled and unlabeled data [34], [35].

The OSVM [36] is a variant of the normal support vector machine (SVM), which works by fitting a hyperplane into the feature space so that it divides the data points into different categories by a largest possible margin [37]. However, with data available from only one class, the OSVM separates the data points from the origin in a high-dimensional transformed feature space constructed using a transformation \( \phi \). The transformed data set is created from the original feature space by a kernel mapping \( k \),

\[
k(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle
\]

that is, the training data that the model receives are the inner products of the samples in the original feature space [36]. For this work, the widely used Radial Basis Function (Gaussian) kernel

\[
k(x_i, x_j) = e^{-\frac{\|x_i-x_j\|^2}{\gamma}}
\]

was chosen, as it is typically the best-performing and even the only viable kernel for one-class problems [38]. A trained OSVM model is described by a plane in the transformed feature space, defined by the support vectors (boundary points) of the training data set. A new data point is classified by evaluating its kernel mapping with the support vectors, thus defining onto which side of the plane the point falls.

**E. OSVM HYPERPARAMETERS**

The complexity of the OSVM decision surface is controlled by two hyperparameters, \( \nu \) and \( \gamma \). The parameter \( \nu \in [0, 1] \) sets both the lower limit for the fraction of the support vectors out of all training samples and the upper limit for the fraction of misclassifications allowed within the training data. On the other hand, \( \gamma > 0 \) controls the kernel bandwidth in (6), that is, the range of influence of the support vectors. In general, increasing \( \nu \) and \( \gamma \) result in a tighter, more complex decision boundary surrounding the target class, whereas smaller values result in a decision boundary that encloses the training data points with a wider margin.

In order to reliably detect damaged capacitors without false alarms, the hyperparameters (\( \nu, \gamma \)) need to be properly tuned. When fitting the OSVM to data from intact MLCCs, the resulting decision surface should enclose the data points compactly enough to detect any anomalous instances. On the other hand, any new data points from the target class should fall within the decision boundary so as to avoid false alarms.

Grid search [39] is a common strategy for hyperparameter optimization: a model is trained and validated over a grid of \( (\nu, \gamma) \) values, and the combination of parameters yielding the best classification performance is selected. However, this approach needs data from both classes, because both positive and negative samples are needed to evaluate the performance of the model. In the context of this study, this would translate into using the intact and damaged MLCC samples both for tuning the model parameters and evaluating the performance. While this could be achieved by means of nested cross-validation, this approach was not chosen as the sample size was limited, and systematic differences between test boards could be present, as discussed in section II-C.

Several methods have been proposed for optimizing the OSVM hyperparameters using data from one class only. Such methods typically apply heuristics-based rules to select the parameters or generate synthetic outliers, which can then be used to optimize the parameters using conventional techniques such as grid search [40]. Many of the proposed approaches, such as [24], [41], [42], optimize for kernel parameter \( \gamma \) only and require determining the value for \( \nu \) beforehand. Some of the more recent approaches, such as [40] and [42], rely on detecting edge patterns, that is, training samples that lie on the edge of the group data points. One such method, based on artificial outlier generation [40], was implemented and tested on the MLCC data. Although the algorithm rivals or outperforms most other algorithms on benchmark data sets, it did not yield unambiguous values for \( \nu \) and \( \gamma \) in the case of the capacitor data set. Instead, several \( (\nu, \gamma) \) pairs with equal preference were discovered. This was likely due to a high ratio of data dimensionality and number of data points, which resulted in nearly all data points being detected as edge patterns. Because the algorithm generates one synthetic outlier per edge pattern, selecting all training data as edge patterns essentially results in overfitting the OSVM model to the training data.
F. PROPOSED HYPERPARAMETER SELECTION ALGORITHM

To optimize the hyperparameter of the OSVM using data from the target class (intact MLCCs) only, a simple heuristic hyperparameter optimization algorithm was designed. Essentially, the algorithm performs a grid search across a plane of \((\nu, \gamma)\) values, evaluating the performance of the OSVM at each grid point to find the optimal hyperparameters. However, in the absence of outlier data (damaged MLCCs), the model is evaluated on target class samples only. Because the training data comprises only dozens of data points, the evaluation is performed by leave-one-out cross-validation, that is, training the model on all but one sample of data, and evaluating the performance on the left-out sample. By repeating this process over each sample in the data set, the mean target class accuracy of the model can be estimated. Fig. 4 illustrates how the cross-validated target class accuracy depends on the choice of the parameters \(\nu\) and \(\gamma\), showing a distinct “plateau” where the model reaches its maximum target class accuracy.

![Fig. 4. Hyperparameters were optimized by mapping the target class accuracy of the OSVM model across a grid of points. At each point, the in-class accuracy of the model was computed using leave-one-out cross-validation. The red dot marks the critical point with the steepest downward curvature in accuracy, defined as the minimum of the Laplacian of the performance plane. The red cross indicates the point of optimal hyperparameters, defined as the point closest to the critical point where the mean accuracy reaches its maximum value, that is, no false alarms are triggered within the training data.](image)

As discussed in Section II-E, the choice of parameters \((\nu, \gamma)\) affects the margin between the OSVM decision surface and the training data points, as well as the complexity of the decision surface. Selecting the points \((\nu, \gamma)\) from the middle of the “plateau” in Fig. 4 will result in a decision surface with a large margin for the training data, which translates into a high target class accuracy. Such a model will rarely, if ever, yield a false alarm on an intact component. However, if the margin between the decision surface and the training data is too large, the model will be insensitive to anomalous instances, such as small signs of mechanical damage in an acoustic signal. Conversely, selecting the hyperparameters such that the target class accuracy is low will result in a model that is oversensitive to outliers and often yields false alarms.

Algorithm 1 \((\nu, \gamma)\) –Performance Surface Mapping

Require: Input data matrix \(X = (x_1 \ldots x_n)^T \in \mathbb{R}^{n \times D}\) (intact MLCCs)  
\#Set parameters for grid mapping 
Set \(n_{grid} \rightarrow 20\)  
Set log-spaced range of \(n_{grid}\) points \(\nu \leftarrow \{10^{-2} \ldots 10^0\}\)  
Set log-spaced range of \(n_{grid}\) points \(\gamma \leftarrow \{10^{-3} \ldots 10^2\}\)  
Preallocate accuracy matrix \(A = (a_{ij}) \in \mathbb{R}^{n_{grid} \times n_{grid}}\)  
\#Partition data for leave-one-out cross-validation 
for \(n = 1 \rightarrow N\) do  
\#Perform grid mapping over hyperparameter space using partitioned data  
Set \((X_{train}, Y_{train}) = (X \setminus \{x_n\}, 0^{(N-1)\times1})\)  
Set \((X_{test}, Y_{test}) = (x_n, 0)\)  
for \(i = 1 \rightarrow n_{grid}\) do  
for \(j = 1 \rightarrow n_{grid}\) do  
Train one-class SVM\((\nu_j, \gamma_j)\) on \((X_{train}, Y_{train})\)  
Evaluate the SVM on \((X_{test}, Y_{test})\)  
Calculate evaluation accuracy \(acc_{\nu, \gamma}\)  
Update \(a_{ij} \leftarrow a_{ij} + acc_{\nu, \gamma}\)  
end for  
end for  
end for  
return \((1/N) \cdot A\)

The fundamental premise of the proposed algorithm is that the optimal hyperparameters \((\nu_{opt}, \gamma_{opt})\) can be found at the edge of the plateau in the hyperparameter space (see Fig. 4). By selecting such a point, 1) the cross-validated accuracy within the target class is close to 100%, and 2) even a small increase in the complexity of the decision plane results in a significant reduction in the target class accuracy. Heuristically, condition 1 ensures that the model gives very few false alarms on target class data, whereas the purpose of condition 2 is to minimize the margin between the decision boundary and the training samples, resulting in a classifier sensitive to outliers.

The algorithm\(^1\) was implemented in two parts. Algorithm 1 constructs a 2D map (as in Fig. 4) of the target class accuracy \(A \in \mathbb{R}^{n_{grid} \times n_{grid}}\) for the OSVM by evaluating the model over a log-spaced grid of hyperparameters \(\nu\) and \(\gamma\). The evaluation is performed using the method of leave-one-out cross validation, which uses as much of the data for training the model as possible. The second part, Algorithm 2, finds a “critical” point \((\nu_c, \gamma_c)\) on the accuracy surface \(A\) where the target class accuracy of the model starts falling most steeply, that is, where the downward curvature of the surface reaches its highest value. This point can be found where the Laplacian

\(^1\)Note that in this study, damaged MLCC samples (outliers) were denoted by the label 1, which is to be understood as a positive test result. However, literature on OSVM algorithms commonly refers to the target class samples as 1 and outliers as 0. The results for the benchmark data sets follow the latter convention.
Algorithm 2 Hyperparameter Selection

Require: Accuracy matrix $A = [a_{ij}]$
Require: Hyperparameter grid $(v, \gamma)$
Require: Threshold switch $s \in \{True, False\}$
Require: Threshold value $T$

Calculate discrete Laplacian $\nabla^2 A$
Find grid coordinates $(i_c, j_c) \leftarrow \arg \min_{i,j} (\nabla^2 A)_{ij}$

if $s = True$ or $A(i_c, j_c) \geq T$ then
    Set $(v_{opt}, \gamma_{opt}) \leftarrow (v_{c}, \gamma_{c})$
else
    Set $(v_{opt}, \gamma_{opt}) \leftarrow \arg \min_{(i,j) \in \arg \max A} \| (i, j) - (i_c, j_c) \|_2$
end if

return $((1 - 1/N)v_{opt}, \gamma_{opt})$

of $A$ reaches its highest negative value:

$$ (v_{c}, \gamma_{c}) = \arg \min_{v, \gamma} (\nabla^2 A). \quad (7) $$

In other words, the algorithm locates the sharpest part of the “cliff” on the accuracy surface, as seen in Fig. 4. Hence, the algorithm was named Cliffhanger.

In order to validate the basic principle of finding the optimal hyperparameters according to (7), the algorithm was tested on several benchmark data sets from the UCI machine learning repository. The test results indicate that such a choice of hyperparameters indeed generally yields classification performance on a par with other hyperparameter selection techniques, with a more in-depth analysis provided in Appendix. However, the good overall performance of the algorithm was achieved at the cost of a number of false alarms because the threshold class accuracy of the model at $(v_{c}, \gamma_{c})$ did not not reach 100%. In some applications, such as detecting damaged MLCCs, avoiding false alarms should be prioritized. For such tasks, an optional threshold condition was imposed on the selection of $v$ and $\gamma$. When using the threshold, Algorithm 2 checks whether the cross-validated accuracy at $(v_{c}, \gamma_{c})$ is above a given threshold value $T$. If the condition is not satisfied, the algorithm locates the point $(v, \gamma)$ closest to $(v_{c}, \gamma_{c})$ in terms of grid coordinates such that the cross-validated accuracy at $(v, \gamma)$ is exceeded. For this study, a strict threshold value of

$$ T = \max(A) \quad (8) $$

was imposed. Such a condition was chosen because in the case of detecting faulty components, false alarms can be considered more costly than false negative results. If a false alarm from a single component results in the rejection of the entire device, the ratio of false alarms should be kept as low as possible. It was also experimentally verified that by using the threshold $T$ instead of $(v_{c}, \gamma_{c})$ yielded significantly less false alarms on pristine MLCCs.

After finding the values for $v_{opt}$ and $\gamma_{opt}$, the output value for $v_{opt}$ in Algorithm 2 is scaled by a factor of $1 - 1/N$. This is done because the parameter $v$ sets the minimum fraction of how many data points within the training set are used as support vectors. As the cross-validation results in Algorithm 1 are obtained with training sets of $N - 1$ points each, the final value for $v_{opt}$ must be rescaled for training data of $N$ points. After optimizing the hyperparameters using Algorithms 1 and 2, the final OSVM model is trained using the full data set.

The functionality of the Cliffhanger algorithm was verified against other OSVM hyperparameter optimization methods on six commonly used data sets from the UCI machine learning repository, with the results listed in Appendix. The algorithm was evaluated both with the threshold (8) (Cliffhanger-T), and without it (Cliffhanger). Based on the results, Cliffhanger generally performs on a par with the reference methods, whereas Cliffhanger-T yields significantly fewer false alarms at the cost of overall performance. However, Cliffhanger-T was selected for this study, as avoiding false alarms is of high priority. While the proposed algorithm has a high computational complexity owing to the combination of grid search and cross-validation, the results in Appendix indicate that it is suitable for situations where the number of samples is low and the ratio D/N is high.

G. EVALUATION METHODS

After optimizing the hyperparameters and training the OSVM on intact capacitors from PCB 2, its classification performance was tested on measurements from PCB 1 both before and after bending. The performance of the model was evaluated using a confusion matrix, which is a common tool for visualizing the output of a classifier model as

$$ \begin{bmatrix} TP & FP \\ FN & TN \end{bmatrix} $$

where TP = True Positive (damaged), FP = False Positive, FN = False Negative, and TN = True Negative (intact). When evaluating the model on intact components only, accuracy (correct outputs/incorrect outputs) was used as the main evaluation metric. When both damaged and nondamaged components were present, precision, recall, and Matthews Correlation Coefficient (MCC) were used as the indicators of performance. In short, precision describes how many of the positive outputs (alarms) are relevant, whereas recall tells how many of the positive samples were discovered by the model. MCC can be seen as a balanced performance score, which accounts for the relative frequencies of each category in the data. Precision, recall, and MCC can be calculated from the data in the confusion matrix as

$$ \text{Precision} \in [0, 1] = \frac{TP}{TP + FP} $$

$$ \text{Recall} \in [0, 1] = \frac{TP}{TP + FN} $$

$$ \text{MCC} \in [-1, 1] = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \quad (10) $$
TABLE 4. OSVM classification results using various performance metrics. Hyperparameter optimization and model training was performed on components from another circuit board (PCB 2). The optimization-training-process was repeated separately on various combinations of input features, which were grouped according to the physical quantity (amplitude, frequency, phase) they represent. The ROC graphs for the combined before/after data are shown in Fig. 5.

| Input features (feat.#) | PCB 1, before bending | PCB 1, after bending | Before & after combined |
|------------------------|-----------------------|---------------------|------------------------|
|                        | Accuracy (%) | Precision | Recall | MCC | AUC-ROC |
| Amplitude | (1,2,6) | 96.67 | 75.00 | 0.8235 | 0.7567 | 0.4865 | 0.6655 | 0.8574 |
| Frequency | (3,4,7) | 30.00 | 43.33 | 0.8000 | 0.1081 | 0.1137 | -0.3879 | 0.2318 |
| Phase | (5,8) | 98.33 | 68.33 | 0.7364 | 0.7568 | 0.3248 | 0.6155 | 0.8460 |
| Amplitude + Freq | (1,2,3,4,6,7) | 43.33 | 78.33 | 0.8333 | 0.8108 | 0.5458 | 0.3081 | 0.6724 |
| Amplitude + Phase | (1,2,5,6,8) | 96.67 | 73.33 | 0.7692 | 0.8108 | 0.4276 | 0.6605 | 0.9049 |
| Freq + Phase | (3,4,5,7,8) | 23.33 | 61.67 | 0.7059 | 0.6487 | 0.2098 | -0.0255 | 0.4865 |
| All | (1-8) | 51.67 | 80.00 | 0.8205 | 0.8649 | 0.5714 | 0.4018 | 0.6923 |

In addition to the metrics derived from the confusion matrix, the performance of the model was evaluated using the Receiver Operating Characteristic (ROC) graph, along with the Area-Under-Curve value (AUC-ROC) [43].

III. RESULTS

After extracting the features in Table 2 and composing the data set according to Table 3, the classification performance of the OSVM was evaluated for two objectives: to discover damaged MLCCs and to avoid false alarms on undamaged capacitors. To this end, the data were divided into training and testing sets based on how the components populated the test PCBs. The data from intact PCB 2 were used for optimizing the hyperparameters using Cliffhanger (Algorithms 1 and 2), after which the OSVM was trained on the same data. The model was then tested on two separate groups of data: intact MLCCs from PCB 1 before the bending procedure, and the same components after the PCB had undergone the bending. This approach was chosen over more commonly used cross-validation techniques, such as the k-fold cross-validation, for two reasons. First, each PCB underwent the acoustic measurements on different occasions, which might translate into data bias even between two sets of intact capacitors, as discussed in Section II-C. Secondly, because the data originated from only two PCBs (PCB 1 was measured twice), this approach allowed for conducting the training and evaluation process for the damaged MLCCs on different individual components, as opposed to using components from PCB 1 for both training and testing the model, potentially causing data leakage.

Ideally, the model should not be able to distinguish the pristine components on PCB 1 from those on PCB 2, whereas components on PCB 1 damaged in the bending process should be detected by their anomalous acoustic signatures. Furthermore, in order to find out which features in the acoustic signal are reliable indicators of damage without being sensitive to changes between intact components, the OSVM was tested on several combinations of features, grouped according to the physical quantity (amplitude, frequency, phase) they represent.

Test results for components from PCB 1 before and after bending are shown in Table 4. The results on intact MLCCs (PCB 1, before bending) show that by using amplitude- and frequency-related information, the model sees little difference between intact components on PCB 1 and those within the training set (PCB 2). However, the inclusion of frequency-related features consistently yielded a significant number of false alarms, making them unfit for the fault detection application using the current experimental setup. However, this does not necessarily mean that the frequency-related features do not carry information about the condition of the component, as the result could be explained by how the frequency-related features are extracted from the peaks in the acoustic signal.

The results for MLCCs on PCB 1 after bending show that the model is able to discover the majority of the damaged capacitors, except when using only frequency-related features. The best classification performance in terms of accuracy, recall, and MCC is obtained using the combination of all eight features, which, however, results in a high error rate on pristine MLCCs. For all feature combinations, the model yields significantly more false alarms on data from PCB 1 after bending compared to pre-bending data, even though the data originate from the same components. This might result from failing to notice some signs of damage during the optical inspection, while another explanation is that the structure of some of the components was affected by the bending procedure, while not resulting in actual damage.

To summarize the results, the model outputs for data before and after the bending procedure were combined. Using MCC as the indicator of performance, the best results were obtained by using amplitude-related features only. The combination of amplitude and phase yielded a higher recall rate than amplitude alone, while also giving more false alarms on the bent PCB. The addition of phase information may have actually led to discovering damaged capacitors on the bent board, which were left undetected during the optical inspection. As the goal would be to avoid false alarms as much as possible while maintaining a decent recall rate, the best feature combinations would thus be either amplitude only, or amplitude combined with phase information. The ROC graphs for the model in Fig. 5 also show that amplitude, phase, and the combination of the two clearly dominate over the rest of
the feature combinations in terms of classification performance, further confirmed by the AUC-ROC scores in Table 4.

The results on amplitude-based features were selected for further analysis due to the highest MCC score on the combined data and good precision on the bent PCB 1, even though the combination of amplitude and phase reached the highest AUC-ROC score. Confusion matrices based on amplitude-based input features in Fig. 6 show that the model fails to detect nine out of 37 damaged MLCCs, while yielding six false alarms. However, only two false alarms are given on the pre-bending data even though the same individual components are in question.

In order to shed more light on the misclassified MLCC samples, the data set was visualized using the t-distributed Stochastic Neighborhood Embedding (t-SNE) algorithm [44]. The algorithm embeds the original, high-dimensional data in a 2-D presentation while attempting to retain the structure of the data set such that the neighboring points in the original space are also close to each other in the 2D embedding. The t-SNE visualization of the entire data set is shown in Fig. 7 along with the OSVM classification results from Fig. 6b. The majority of the capacitors labeled as damaged constitute a cluster separate from the intact samples, suggesting that mechanical damage affects the acoustic signature of an MLCC in a consistent manner. The way in which the OSVM specifically classifies the capacitors in this cluster as anomalous further supports this observation. However, five or six samples within this cluster appear to be false positives. In contrast, seven to ten out of 37 samples labeled as “damaged” are misclassified by the OSVM, and the majority of these samples appear to lie well within the group of intact capacitor samples.

Given that the capacitors were manually labeled by visually inspecting the cross-sectional and X-ray images, it is likely that some damage was left unseen. In contrast, artifacts such as scratches in the epoxy may have led to labeling of some undamaged MLCCs as damaged. To further verify the inspection results, the cross-sectional images from the six false positive and nine false negative samples in Fig. 6b were reinspected. Two false positive samples indeed showed a small crack in the lower corner of the capacitor body, and two others showed potential signs of delamination between the end termination and the solder joint. As for the false negatives, only two out of seven definitely showed cracks, and these findings were also confirmed by X-ray. Possible small cracks were seen in three other samples, and the cracks observed previously in the rest of the samples were probably scratches in the epoxy surface of the cross-section samples.

FIGURE 5. ROC graph describing the performance of the OSVM model using different feature groups on the combination of data from before and after bending PCB 1. The Area-Under-Curve scores are listed in Table 4.

FIGURE 6. Confusion matrices for PCB 1 before and after bending the circuit board. The model was trained using only amplitude-related features, as this feature set yielded the highest MCC score on the combination of pre- and post-bending data. Class 1 = damaged component, class 0 = no damage.

FIGURE 7. t-SNE visualization of the acoustic data set, showing the eight-dimensional data embedded in two dimensions. MLCCs from intact PCBs 1 and 2 are indicated by gray, whereas the red and blue markers indicate the microsectioning and X-ray inspection results for the capacitors on the bent PCB 1. The ring symbols indicate the capacitors marked by the OSVM (using amplitude features only) as anomalous.
alarms rate of approx. 4% can still be considered too high for selecting the parameters of the OSVM. Even though the false parameter optimization algorithm performs successfully in a real-life situation is much lower than in this study. Very costly, whereas the occurrence of faulty components of capacitors per device can be high, false alarms can become prioritized over a high ratio of true positives. As the number of damaged components, a low ratio of false alarms should be the data are clustered in the t-SNE visualization. The model was trained using only amplitude-related features.

IV. DISCUSSION

Acoustic characterization is a promising method for nondestructive testing of MLCCs, especially because it can be performed on soldered components on assembled circuit boards. However, in order to generalize the method to capacitors of various sizes and capacitances, it should be possible to detect damaged components based on pristine examples only. The results of this study show that by using a one-class SVM in conjunction with the proposed combination of preprocessing, feature extraction, and hyperparameter optimization, 75–80% of damaged MLCCs can be detected, while maintaining a false detection rate below 4% on pristine components. However, the performance values on damaged capacitors contain some uncertainty, as the reinspection of the component samples revealed that eight out of fifteen misclassified capacitors were actually mislabeled. Thus, the method can be expected to reach a detection rate of over 90% if a larger data set with more accurate labels is available. Nevertheless, the classification performance on the bent PCB I is similar to that of the binary SVM in [23], while the OSVM maintains a low false alarm rate on the intact PCB was also over 50%. This result can be explained by the way in which the feature values are distributed: even though some acoustic features may be strongly correlated with structural damage in MLCCs, intrinsic deviations in the physical parameters of the components and environmental factors will cause feature values from two different populations of intact components to overlap. By appropriate feature selection, a nonoverlapping region between the two classes can be found.

This study built upon prior research [23], where the use of resonant frequencies $A_1$ and $A_2$ was evaluated as input features for an ordinary two-class SVM, reporting an accuracy of 78.3%. Because the study was only performed on capacitors on a single bent PCB assembly (PCB 1), no information of the performance on pristine capacitors is available from the study. Nevertheless, the classification performance of the one-class SVM on the bent PCB I is similar to that of the binary SVM in [23], while the OSVM maintains a low false alarm rate on the same exact components prior to PCB bending. While these observations confirm that the proposed feature extraction and hyperparameter optimization methods are applicable for the task of detecting faulty MLCCs, they also further suggest that the performance results may be at least partially limited by the labeling process.

A comprehensive comparison with other proposed damage detection approaches is difficult as there is only limited information available on the accuracy of the other methods. However, the insulation resistance measurement, a common quality assurance tool, is typically capable of revealing only a minority of cracks [4]. The electromechanical resonance spectrum measurement [19] has been suggested to be an accurate method for larger cracks, but less sensitive for instance to small thermal cracks, and it is also affected by the prehistory of the component [4]. Furthermore, the method might be impractical for production PCB assemblies, as it requires electrical measurements, which are easily affected by other components on the assembly. In contrast, the acoustic method used in this study can be isolated to a single capacitor even if there are parallel components on the path of the excitation signal, given that the excitation voltage source has a sufficiently low source impedance.

While the proposed method performs successfully in detecting damaged MLCCs, the accuracy should be verified and improved using a larger number of capacitors. Even though this study was performed on capacitors of only a single case size, prior studies [22], [23], [28] have shown that MLCCs of other case sizes exhibit similar changes in acoustic behavior when mechanically damaged, suggesting that the method should be applicable to other case sizes as well. Moreover, the higher amplitude of acoustic emissions observed in smaller case sizes [22] will likely result in a better
signal-to-noise ratio, making the acoustic identification of damage easier for smaller MLCCs. However, validating this method on MLCCs of other case sizes and capacitances is a matter of further study. In addition, component sample misclassifications should be reduced by incorporating other labeling methods, such as the C-scan ultrasonic microscopy. On the other hand, the performance may vary with different capacitor sizes as well as types and degrees of physical damage, and thus, further study is required. Nonetheless, the proposed method performs well and could be a valuable tool for end-of-line testing of PCB assemblies with large MLCCs, which are especially prone to flex cracking. As the experiments were performed on data from custom-built test boards, the method should also be tested on a production PCB assembly.

V. CONCLUSION
Cracks and delaminations in MLCCs related to soldering, PCB assembly, and handling remain problematic, as there are no commonly used tools for fast and reliable screening for these types of damage. In this work, an acoustic nondestructive testing method for MLCCs was demonstrated on assembled PCBs. For detecting damaged or anomalous capacitors, the method employs a one-class support vector machine, a machine learning model trained solely on pristine capacitor examples. Furthermore, an algorithm for optimizing the hyperparameters of the model is presented.

First, acoustic measurement data from MLCCs was obtained from a prior study. The data contained measurements from 120 intact capacitors on two test PCBs, and another 60 measurements after one of the boards was subjected to controlled bending. The measurement data were preprocessed and composed into a data set, with each bent sample labeled as damaged or nondamaged according to cross-sectional and X-ray images. An OSVM model was then optimized and trained on data from intact capacitors on one PCB, and the classification accuracy was tested with MLCCs from another PCB before and after bending. The results show that through proper feature engineering and hyperparameter optimization, the model is capable of successfully identifying damaged MLCCs with dozens of intact capacitors as training data, while maintaining a low rate of false alarms on pristine capacitors.

APPENDIX

TEST RESULTS ON UCI MACHINE LEARNING DATA SETS
The proposed hyperparameter selection algorithm was tested on eight commonly used data sets from the UCI machine learning repository: Breast Cancer Wisconsin (Diagnostic), Heart Disease, Pima Indians Diabetes, Connectionist Bench (Sonar, Mines vs. Rocks), Wine, Glass Identification, Statlog (Vehicle Silhouettes), and Connectionist Bench (Vowel Recognition - Deterding Data). The data sets have previously been used for testing one-class support vector machine hyperparameter optimization in [24] and [42], and a summary of the data sets is shown in Table 5.

The performance of the proposed algorithm was compared against other hyperparameter optimization algorithms, following the procedure in [42]. The data sets were first preprocessed by removing rows with missing values, as well as removing all-constant columns. The feature variables were then standardized to zero mean and unit variance. For each data set, the target class was selected as in [42], and randomly selected 80% of the target samples were used for hyperparameter optimization and training the model. The model was then tested on the outlier instances, plus the remaining 20% of the target data. As in [42], the evaluation process was repeated 20 times, randomly partitioning the target data each time. The performance of the model was evaluated using the geometric mean (G-mean) of True Positive Rate (TPR, i.e., Recall) and True Negative Rate (TNR):

$$G\text{-mean} = \sqrt{\text{TPR} \cdot \text{TNR}}$$

The G-mean scores of the Cliffhanger algorithm were compared against those of five other algorithms, as reported in [42]: MIES [42], Min#SV+MaxL (MSML) [45], SKEW [46], VM [47] and MD [48]. Two versions of the Cliffhanger algorithm are compared with the reference methods: Cliffhanger, which selects the point \((v, \gamma)\) as the optimum hyperparameters; and Cliffhanger-T, the version used for the MLCC data which imposes an additional condition \((v, \gamma) \in \text{arg max} A\) for the hyperparameters. The performance of these algorithms in terms of the G-mean score, as reported in [42] is listed in Table 6. To highlight the differences between Cliffhanger and the thresholded version, Cliffhanger-T, the raw confusion matrix elements (refer to (9)) are given in Table 7.

The results in Table 6 show that the performance of the Cliffhanger algorithm is generally on a par with the reference methods. In terms of G-mean scores, Cliffhanger yielded better results than the reference methods on three out of eight data sets. On the other hand, Cliffhanger-T, which imposes a more strict requirement for the hyperparameters, performs comparably with other methods only on Cancer, Wine, Glass, and Vowel data sets. The raw confusion matrix values for both variants of the Cliffhanger algorithm in Table 7 shows that Cliffhanger-T has a significantly lower rate of false alarms (FPR) that the non-thresholded version.

### Table 5. UCI ML repository data sets.

| Data set       | Dimensions | Targets | Outliers |
|----------------|------------|---------|----------|
| Cancer         | 9          | 458     | 241      |
| Heart          | 13         | 164     | 139      |
| Diabetes       | 8          | 500     | 268      |
| Sonar          | 60         | 111     | 97       |
| Wine           | 13         | 48      | 130      |
| Glass          | 9          | 70      | 144      |
| Vehicle        | 18         | 199     | 647      |
| Vowel          | 10         | 48      | 480      |

2https://archive.ics.uci.edu/ml/datasets.php
of the algorithm, at the cost of the ability of discovering positive instances. This makes the Cliffhanger-T better suited for applications where some missed positive instances are acceptable, but false alarms are very costly.

It must be noted that the performance of the Cliffhanger algorithm ultimately depends on the given data: if the plateau of maximal in-class accuracy is close to the point \((\nu_c, \gamma_c)\) (see Fig. 4), the resulting model is more sensitive for detecting outliers than if the plateau is further away from \((\nu_c, \gamma_c)\). For MLCC data, the plateau extends close to \((\nu_c, \gamma_c)\), in which case it is justifiable to impose stricter requirements for the hyperparameters. All in all, the Cliffhanger algorithm performs adequately when compared with the results reported by [42]. However, the authors would like to stress that the results for the reference methods in Table 6 were obtained from another study. While the testing methodology of the study was followed as precisely as possible, there might still be differences between data preprocessing, splitting, and selection. Therefore, these results should be viewed as general indicators of the performance of Cliffhanger, and a deeper analysis of the algorithm is a matter of further study.

REFERENCES

[1] R. Randoll, W. Wondrak, and A. Schletz, “Lifetime and manufacturability of integrated power electronics,” Microelectron. Rel., vol. 64, pp. 513–518, Sep. 2016.
[2] B. Sood. (Mar. 2013). Root Cause Failure Analysis of Electronics. University Of Maryland. [Online]. Available: https://www.smta.org/chapters/files/Philadelphia_SMTA_Philly_Rev2.pdf
[3] J. Ho, T. R. Jow, and S. Boggs, “Historical introduction to capacitor technology,” IEEE Elect. Insul. Mag., vol. 26, no. 1, pp. 20–25, Jan. 2010.
[4] A. Teverovsky, “Cracking problems in low-voltage chip ceramic capacitors,” NASA, Washington, DC, USA, Tech. Rep. GSFC-E-DAA-TN65668, Mar. 2018. [Online]. Available: https://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/20190001592.pdf
[5] D. Brown, “Oxygen vacancy migration in MLCCs,” in Proc. Pan Pacific Microelectron. Symp. (Pan Pacific), Feb. 2018, pp. 1–6.
[6] J. Al Ahmar and S. Wiese, “A finite element modelling and fracture mechanical approach of multilayer ceramic capacitors,” in Proc. 16th Int. Conf. Thermal, Mech. Multi-Phys. Simulation Exp. Microelectron. Microsys., Apr. 2015.
[7] M. Keimasi, M. H. Azarian, and M. G. Pecht, “Flex cracking of multilayer ceramic capacitors assembled with pb-free and Tin–Lead solders,” IEEE Trans. Device Mater. Rel., vol. 8, no. 1, pp. 182–192, Mar. 2008.
[8] D. Han, “Material shortage & price trend,” Silecta, Legget Drive, ON, Canada, Tech. Rep., Feb. 2018, pp. 206–308. Accessed: Feb. 22, 2019. [Online]. Available: https://scmroundtable.com/wp-content/uploads/2018/03/material-shortage-price-trend-report.pdf
[9] D. M. Zoghi. (Jan. 2018). MLCC Shortages by Case Size: 2018. TTI. Accessed: Feb. 22, 2019. [Online]. Available: https://www.tiine.com/content/tiine/en/resources/marketeeye/categories/passives/me-zoghi-20180930.html
[10] C. Andersson, O. Kristensen, E. Varescon, and F. Iannuzzo, “Impact of bending speed and setup on flex cracks in multilayer ceramic capacitors,” in Proc. IEEE 11th Int. Symp. Diag. Electr. Mach., Power Electron. Drives (SDEMPED), Aug. 2017, pp. 608–614.
[11] S. R. Kahn and R. W. Checkaneck, “Acoustic emission testing of multilayer ceramic capacitors,” IEEE Trans. Compon. Hybrids Manuf. Technol., vol. CHMT-6, no. 4, pp. 517–526, Dec. 1983.
[12] C. Andersson, O. Kristensen, S. Miller, T. Gloor, and F. Iannuzzo, “Lock-in thermography failure detection on multilayer ceramic capacitors after flex cracking and temperature–humidity–bias stress,” IEEE J. Emerg. Sel. Topics Power Electron., vol. 6, no. 4, pp. 2254–2261, Dec. 2018.
[13] D. N. Donahoe, M. Pecht, I. K. Lloyd, and S. Ganesan, “Moisture induced degradation of multilayer ceramic capacitors,” Microelectron. Rel., vol. 46, no. 2, pp. 400–408, 2006. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0026271405001022

[14] T. Adams, “Preventing MLCC failures,” Circuits Assem., vol. 6, p. 17, Aug. 2000. [Online]. Available: http://www.sonosanchnica.cn/feat/images/Preventing-MLCC-Failures-Circuits-Assembly-August-09.pdf

[15] G. Vogel, “Avoiding flex cracks in ceramic capacitors: Analytical tool for a reliable failure analysis and guideline for positioning cercaps on PCBs,” Microelectron. Rel., vol. 55, nos. 9–10, pp. 2159–2164, 2015.

[16] C. Andersson, J. Ingman, E. Varescon, and M. Kiviniemi, “Detection of cracks in multilayer ceramic capacitors by X-ray imaging,” Microelectron. Rel., vol. 57, no. 6, pp. 352–356, Sep. 2016.

[17] J. M. Ingman, J. P. A. Jormaainainen, A. M. Vulli, J. D. Ingman, K. Maura, T. J. Kärkkäinen, and P. Silventoinen, “Localization of dielectric breakdown defects in multilayer ceramic capacitors using 3D X-ray imaging,” J. Eur. Ceram. Soc., vol. 39, no. 4, pp. 1178–1185, Apr. 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0955221918306617

[18] B. H. Ko, S.-G. Jeong, Y.-G. Ahn, K.-S. Park, N.-C. Park, and Y.-P. Park, “Analysis of the correlation between acoustic noise and vibration generated by a multi-layer ceramic capacitor,” Microsyst. Technol., vol. 20, nos. 8–9, pp. 1671–1677, Aug. 2014, doi: 10.1007/s00542-014-2209-5.

[19] L. Bechou, S. Mejdi, Y. Ousten, and Y. Danto, “Non-destructive detection and localization of defects in multilayer ceramic chip capacitors using electromechanical resonances,” Qual. Rel. Eng. Int., vol. 12, no. 1, pp. 43–53, Jan. 1996.

[20] W. L. Johnson, S. A. Kim, G. S. White, and J. Herzberger, “Nonlinear resonant acoustic detection of cracks in multilayer ceramic capacitors,” in Proc. IEEE Int. Ultrason. Symp., Sep. 2014, pp. 244–247.

[21] W. L. Johnson, J. L. Herzberger, S. A. Kim, K. L. Peterson, P. R. Heyliger, and G. S. White, “Resonant acoustic frequency shifts associated with cracks in multilayer ceramic capacitors,” IEEE Trans. Device Mater. Rel., vol. 17, no. 2, pp. 316–323, Jun. 2017.

[22] S. Levikari, T. J. Kärkkäinen, C. Andersson, J. Tamminen, and P. Silventoinen, “Acoustic phenomena in damaged ceramic capacitors,” IEEE Trans. Ind. Electron., vol. 65, no. 1, pp. 570–577, Jan. 2018.

[23] S. Levikari, T. J. Kärkkäinen, C. Andersson, J. Tamminen, and P. Silventoinen, “Acoustic detection of cracks and delamination in multilayer ceramic capacitors,” IEEE Trans. Ind. Appl., vol. 55, no. 2, pp. 1787–1794, Mar. 2019.

[24] A. Anaisi, N. L. D. Khoa, and Y. Wang, “Automated parameter tuning in one-class support vector machine: An application for damage detection,” Int. J. Data Sci. Anal., vol. 6, no. 4, pp. 311–325, Dec. 2018, doi: 10.1007/s41060-018-0151-9.

[25] S. Das, A. N. Srivastava, and A. Chattopadhyay, “Classification of damage signatures in composite plates using one-class SVMs,” in Proc. IEEE Aeros. Conf., Mar. 2007, pp. 1–19, doi: 10.1109/AERO.2007.352912.

[26] K. Yan, Z. Ji, and W. Shen, “Online fault detection methods for chillers combining extended Kalman filter and recursive one-class SVM,” Neurocomputing, vol. 228, pp. 205–212, Mar. 2017.

[27] B. M. Mun, M. Lim, and S. J. Bae, “Condition monitoring scheme via one-class support vector machine and multivariate control charts,” J. Mech. Sci. Technol., vol. 34, no. 3, pp. 3937–3944, Oct. 2020.

[28] S. Levikari, “Detection of cracks: Acoustic experiments on multilayer ceramic capacitors,” M.S. thesis, LUT School Energy Syst., Lappeenranta Univ. Technol., Lappeenranta, Finland, 2018.

[29] S. W. Smith, B. M. Mun, M. Lim, and S. J. Bae, “Survey of outlier detection methodologies,” Artif. Intell. Rev., vol. 22, no. 2, pp. 85–126, 2004. [Online]. Available: https://doi.org/10.1007/s10462-004-0430-y

[30] V. Chandola, A. Banerjee, and V. Kumar, “Anomaly detection: A survey,” ACM Comput. Surv., vol. 41, no. 3, pp. 1–58, Jul. 2009, doi: 10.1145/141880.141882.

[31] X. Zhu and A. B. Goldberg, “Introduction to Semi-Supervised Learning,” Morgan & Claypool: San Rafael, CA, USA, 2009. [Online]. Available: https://doi.org/10.2200/S00196ED1V1Y200906AIM006

[32] I. Triguero, S. García, and F. Herrera, “Self-labeled techniques for semi-supervised learning: Taxonomy, software and empirical study,” Knowl. Inf. Syst., vol. 42, no. 2, pp. 245–284, Feb. 2015, doi: 10.1007/s10115-013-0706-y.

[33] B. Scholkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson, “Estimating the support of a high-dimensional distribution,” Neural Comput., vol. 13, no. 7, pp. 1443–1471, Jul. 2001.

[34] A. Bousnier and G. M. Madden, “Kernels for one-class support vector machines,” in Proc. Int. Conf. Inf. Sci. Appl. (ICISA), May 2014, pp. 1–4.

[35] P. Silventoinen, “Acoustic phenomena in damaged ceramic capacitors,” IEEE Trans. Ind. Appl., vol. 65, no. 1, pp. 570–577, Jan. 2018.

[36] P. Silventoinen, “Acoustic detection of cracks in multilayer ceramic capacitors,” IEEE Trans. Ind. Appl., vol. 45, no. 2, pp. 400–408, 2006. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0018931605001022

[37] C. Cortes and V. Vapnik, “Support-vector networks,” Machine Learn., vol. 20, no. 3, pp. 273–297, Sep. 1995, doi: 10.1007/BF00994018.

[38] S. A. Kim, W. L. Johnson, and J. Herzberger, “Resonant acoustic frequency shifts associated with microcracks in ceramic capacitors,” in Proc. IEEE Int. Conf. Ind. Electron. (ICIE), Sep. 2014, pp. 244–247.

[39] B. M. Mun, M. Lim, and S. J. Bae, “Condition monitoring scheme via one-class support vector machine,” in Proc. Int. Conf. Inf. Sci. Appl. (ICISA), May 2014, pp. 1–4.
CAROLINE ANDERSSON (Member, IEEE) was born in Sweden, in 1983. She received the M.Sc. degree from the Faculty of Engineering, Lund University (LTH), Sweden, and the Ph.D. degree from the Swiss Federal Institute of Technology (ETH), Zurich, Switzerland. She is currently working as a Senior Research and Development Engineer with ABB Traction, Switzerland, on the topic of reliability of power electronics. Her main research interests include accelerated life testing, physics of failure, and reliability of components and systems.

JUHA TAMMINEN (Member, IEEE) was born in Finland, in 1980. He received the B.Eng. degree in electronics from the Espoo–Vantaa University of Applied Sciences, in 2006. He is currently working as the Design Manager with ABB Drives, Finland, in production testing. His work concentrates on improving production testing concepts in pursuit of better quality and productivity.

MIKKO NYKYRI (Member, IEEE) was born in Finland, in 1993. He received the B.Sc. and M.Sc. degrees from the Lappeenranta University of Technology (LUT), in 2017 and 2018, respectively. He is currently pursuing the Ph.D. degree with the Laboratory of Applied Electronics, LUT School of Energy Systems. His main research interests include utilization of data analytics, machine learning, and the IoT in industrial predictive maintenance.

PERTTI SILVENTOINEN (Member, IEEE) was born in Simpele, Finland, in 1965. He received the D.Sc. degree from the Lappeenranta University of Technology (LUT), Lappeenranta, Finland, in 2001. He became a Professor of applied electronics in 2004. His current research interest includes power electronics systems in various applications.

***