Acoustic feature based geometric defect identification in wire arc additive manufacturing

Nowrin Akter Surovi & Gim Song Soh

Published online: 16 May 2023.

Submit your article to this journal ⍌

Article views: 159

View related articles ⍌

View Crossmark data ⍌

Taylor & Francis
Taylor & Francis Group

Acoustic feature based geometric defect identification in wire arc additive manufacturing

Nowrin Akter Surovi [ORCID] and Gim Song Soh [ORCID]

Singapore University of Technology and Design (SUTD), Singapore, Singapore

ABSTRACT

In additive manufacturing of metals, numerous techniques have been employed to sense print defects. Among these, acoustic-based sensing has the advantage of low cost and shows the most potential to identify both external and internal defects as an in-situ monitoring system. Using acoustic signals, researchers have broadly investigated non-machine learning and machine learning-based approaches to identify defects like balling, micro defects, lack of fusion pores, keyhole pores, cracks, and porosity. While most of these works have shown promising results for laser-based AM systems, few have explored how acoustic signals can be used effectively for Wire Arc Additive Manufacturing (WAAM) defect detection. This paper proposes a methodology to construct machine learning (ML)-based models on identifying geometrically defective bead segments using acoustic signals during the WAAM process. Geometrically defective bead segment or geometric defect is a defect that causes voids in the final printed part due to incomplete fusion between two non-uniform overlapping bead segments. Such a defect is currently not explored in the literature. The proposed methodology uses a novel dataset labeling approach to identify good and bad bead segments based on an optimal threshold of the range of mean curvature. Furthermore, the methodology targets defective bead segments based on acoustic feature inputs like Principal Components (PC) or Mel Frequency Cepstral Coefficients (MFCC). To understand the resulting performance of the defect identification models constructed based on the proposed methodology, experiments are performed and tested on a variety of ML models (KNN, SVM, RF, NN, and CNN) based on the Inconel 718 material. The results show that the combinatorics of two acoustic input features and five ML models can be able to identify geometrically defective segments accurately with F1 score that ranges from 80% to 85%.

1. Introduction

Wire Arc Additive Manufacturing (WAAM) is a direct energy deposition process according to ASTM F2792-12a where weld beads are deposited layer-by-layer to form 3D metallic components (Busachi et al. 2015; Ding et al. 2015). WAAM employs an electric arc as a heat source and a metal wire as a feedstock for material deposition (Rodrigues et al. 2019). WAAM is recently becoming popular because of its low equipment cost, low buy-to-fly ratio, high deposition rate, and environmentally friendly approach. However, WAAM suffers from processing-related defects such as porosity, cracks, distortion, oxidation, etc. Xu et al. (2018), Yuan et al. (2020), Wu et al. (2018) and Xia et al. (2020). Sometimes these defects can propagate to the subsequent layers, which reduces the strength (Rajashekar and Rajaprakash 2016; Surovi and Soh 2022) of the final printed

part, shortens the product's lifetime, and sometimes cause a collapse of the structure that causes economic loss (Chu and Wang 2016; Vilček et al. 2017). Therefore, it is important to identify print defects as early as possible to take appropriate corrective measures during the printing process to save welding resources and material costs (Surovi, Dharmawan, and Soh. 2021).

Researchers used various non-destructive techniques (NDT) to identify defects, such as laser-based profile scanning, image-based sensing, thermography, acoustic-based sensing, etc. For instance, on laser-based profile scanning, Chen et al. (2021) used a laser profiler, and Huang et al. (2022) used a 3D laser profilometer inspection (3D-LPI) system for the identification of surface defects. Laser-based profilers identify such defects by capturing the reflected laser line from the target surface through a narrow bandpass optical filter.

However, the limitation of this approach is that ambient light may blend with the reflected laser, particularly on metallic surfaces, affecting its accuracy, and it is incapable of identifying internal defects. On image-based sensing, Li et al. (2023) used a CMOS camera and an optical filter for surface defects detection. Similarly, Cho et al. (2022) used a high dynamic range (HDR) camera to identify normal and abnormal beads. In both approaches, images from the object were analysed for defects. The downside of such an approach is that defects can only be identified if visible on the image, and such sensors require high maintenance. For thermography, Kryukov et al. (2014) used a thermal camera to identify defects by studying the intensity profile during the friction weld process. In his approach, internal defects were identified by monitoring the heat dissipation from the object's cooling rate and thermal gradients. However, additional post-processing is required to identify defects due to a wormhole or lack of fusion. In acoustic-based sensing, the acoustic signal generated during the deposition process is filtered and analysed for features using signal processing techniques to detect defects. They have several advantages over the previously discussed sensors due to their low cost, ease of maintenance, simple structure, and highly targeted sensitivity (Kaushik, Nance, and Ahuja 2005), on top of their potential to identify both external and internal defects.

Several works have analysed the acoustic signal from the electric arc to determine the stability of the deposition process. Typically, researchers use acoustic emission (AE) or acoustic sensors to characterise the process stability, metal transfer modes, or defects. AE sensors are attached to the printing surface and used to capture the sound signal waveform through the material during the deposition process. On the other hand, acoustic sensors are mounted near the printing surface and measure the sound pressure waveform in the air. In the context of using acoustic-based sensing to determine process stability and metal transfer mode, Polajnar, Bergant, and Grum (2013) showed that irregularities in the bead geometry are reflected in the intensity of acoustic and current signals. They also found that acoustic signals with high-frequency arc oscillations indicate process instability. Pal, Bhattacharya, and Pal (2010) studied the relationship of welding arc signals with process parameters and metal transfer mode for both continuous gas metal arc welding (GMAW) and pulsed gas metal arc welding (P-GMAW). They found that the RMS and Kurtosis of arc sound can effectively identify metal transfer modes. Grad et al. (2004) investigated acoustic signals and weld process characteristics for GMAW. They found that non-uniform and discontinuous gas metal arc welds can be associated with non-regularities in their arc behaviour, such as long silence between two successive arc re-ignitions. Lv et al. (2014) showed that the height of the arcs constituting the shape of the bead has a linear relationship with the pressure of the measured acoustic signal. The research mentioned above highlighted that recorded acoustic signal contains rich information on the deposition process and can be used to infer the arc conditions, melt pool dynamics, and even internal properties of the welded material.

In the context of defect detection using acoustic signals, researchers have broadly investigated non-machine learning (non-ML) or machine learning (ML) based approaches to identify different types of defects that occurred in metal AM. Among the non-ML-based approaches, Gaja and Liou (2017) showed that porosity and crack defects of the laser metal deposition (LMD) process could be identified by acoustic signals. They concluded that porosity defect produces acoustic profiles with shorter decay time and less amplitude, while cracks defect triggers acoustic characteristics with short duration and high amplitudes. Ito et al. (2021) showed that AE sensors could pick out the pore location and micro defects occurrence in selective laser melting (SLM). They compared their finding with X-ray computed tomography (CT) and concluded that burst-type AE events correspond to pore and micro defects. Rao et al. (2022) proposed a two-stage ultrasonic waveform tomographic method to detect and characterise porosity defects in multi-material (MM) laser metal deposition (LMD) parts. They concluded that their method could accurately reconstruct the multi-material interface and porosity by lowering the deposition laser power. Recently, Bevans et al. (2023) used a wavelet-integrated graph theory to filter, process, and monitor the acoustic signatures for WAAM. This approach utilised one type of feature, which they term Fiedler number, to pick out defect location accurately. However, it could not classify the specific type of defect that occurs at that location.

On the use of ML approaches for defect detection using acoustic signals, Drissi-Daoudi et al. (2022) explored acoustic features from time, frequency, and time-frequency domains. They used a Convolutional Neural Network (CNN) to identify the lack of fusion pores, conduction mode, and keyhole pores defects for three metallic alloys. They found that the classification of acoustic features is material and defect dependent. They also found that the performance of their process depends on the geometry of the chamber material and the position of the AE sensor. Subsequently, Drissi-Daoudi et al. (2023) developed a robust acoustic-based CNN model to classify the same defects based on spectrograms, where they demonstrated that such defects could be picked out even when caused by other

process parameters that were not used in the training process. They also found that the number of parameter sets, the relative normalised distance between the training and testing set, and the position of the training sets in the process map influence the resulting classification accuracy. Similarly, Chen et al. (2023) researched acoustic features such as time, frequency, and time-frequency domain with different traditional machine learning and CNN models to identify crack and keyhole defects in laser direct energy deposition (LDED). They concluded that the Mel Frequency Cepstral Coefficient (MFCC) - CNN model outperformed all the classic ML models in their experiments. However, their approach required longer acoustic signals for defect detection than others in the literature. Wasmer et al. (2023) compared the performance of seven supervised ML algorithms to identify the lack of fusion and conduction modes of functionally graded materials based on time and frequency domain features. They found that the resulting classification accuracy was low, at around 73%, due to noise created by the gas flow surrounding the laser deposition process zones. Shevchik et al. (2019) combined a Fiber Bragg Grating (FBG) as an AE sensor with a spectral convolutional neural network to classify three quality categories based on porosity. Pandiyan et al. (2021) proposed a semi-supervised approach where two generative Convolutional Neural Networks based on Variational Auto-Encoder, and General Adversarial Networks were used to differentiate balling, lack of fusion (LOF) pores, and keyhole pores during the conduction mode melting process for LPBF. However, their model cannot be generalised across different powder distributions, process parameters, and compositions. Tempelman et al. (2022) extracted time and frequency series features from acoustic signals using Support Vector Machines (SVM) to classify keyhole pores defects in the LPBF process. Their approach identified the keyhole location using an X-ray measurement and correlated it with an acoustic signal. They found that high-frequency signals are an indication of keyhole defect formation. Recently, Kononenko et al. (2023) analysed the AE signal in the LPBF process and, based on a certain threshold with their proposed ML model, separated the AE signature of crack from noise. However, in practical application, if the noise level is high, the threshold would be high, which limits such an approach. A summary of the investigated defects and their limitations for the above-mentioned ML and non-ML papers are shown in Table 1.

The research, as mentioned above, highlighted that time domain, frequency domain, and wavelet analysis are popular techniques to process acoustic signals to determine defects. Specifically, features such as peak amplitude, kurtosis, energy, number of counts, duration,

Table 1. Prior work concerning acoustic defect detection in metal AM.

AM process	Targeted Defects	Limitations
LPBF, LDED, PBF, SLM	Balling, Micro defects, Lack of fusion pores, Keyhole pores, Crack, Porosity	No geometric defect detection
WAAM	Line width variation, Voids, Porosity, Spatter	Not classify specific defects Not scalable to other defects

rise, peak amplitude, frequency, and Mel Frequency Cepstral Coefficient have been used to identify various defects (Surovi, Dharmawan, and Soh. 2021; Surovi, Hussain, and Soh 2022). However, while most of the published works of both ML and non-ML-based approaches show promising results for in-situ monitoring systems in laser-based AM applications, few have explored how acoustic signals can be used effectively for WAAM defect detection. Furthermore, when comparing ML and non-ML-based approaches, we believe that the ML approaches have benefits over the non-ML approaches because they can be scaled easily using a unified framework to include more defect categories if appropriate data are obtained. They also show the potential to yield a higher accuracy because they can consider multiple features in their model, unlike non-ML-based approaches, where only a single feature is considered. This is because non-ML-based approaches often require domain-expert knowledge in the form of hand-crafted rules involving process parameters. Defect detection algorithms using such hand-crafted rules are highly sensitive to the rules. For instance, new rules need to be adopted when introducing a new material or part geometry (Goh, Sing, and Yeong 2021). On the other hand, an ML-based approach (e.g. neural networks) can learn the rules as long as sufficient and accurate data is provided. However, the accuracy of an ML model depends on how accurately the data set is labeled for training. Currently, the datasets reported in the literature are labeled based on visual inspection, which is subjective and error-prone.

In this paper, a methodology for constructing ML-based models to identify geometrically defective bead segments using acoustic sensing during the WAAM process is proposed. This approach targets geometric defects, a defect unique to the WAAM process, due to its sensitivity to process variation. A bead produced using a constant WAAM process is known to produce segments of varying geometrical shapes due to unaccountable process variation. Such defects are not explored in literature as the defects do not cause issues to a single bead segment element itself. However, it has implications

when two overlapping bead segments are required to fuse. It contributes to voids in the final printed part due to the incomplete fusion between two overlapping bead segments, a common problem when performing multi-bead prints as shown in Figure 1.

The ML-based construction methodology proposed in this paper leverages a novel approach toward geometric defect detection through bead segment monitoring and dataset labeling to separate good and bad bead segments based on an optimal threshold of the range of mean curvature. By discretizing defect detection, it has the benefit of identifying a localised defect more accurately so that early intervention can be possible. Currently, labeling of AM defects is highly manual and is based on a visual inspection approach. This is error-prone and time-consuming, especially when the dataset is large. Here an approach to determine an optimal threshold to separate good and bad segments based on overlapping areas of KDE distribution is proposed. To understand the resulting performance of the defect detection model constructed based on the proposed methodology, experiments are performed and tested based on Inconel 718 material.

2. Acoustic signal feature extraction, labeling and defect identification models

Figure 2 shows the methodology of constructing the geometric defect detection model. The approach consists of three elements: acoustic feature extraction, dataset labeling, and defect detection modeling. Two feature extraction techniques: the Principal Component Analysis (PCA) and Mel Frequency Cepstral Coefficients (MFCCs) are explored for their suitability in identifying geometric defects due to their performance in solving other domain-specific problems (Palaz and Collobert 2015; Zhao et al. 2010). The data set labeling uses a

curvature-based labeling approach that heuristically finds a range curvature threshold to separate good and bad segments. Subsequently, the acoustic features and labels are used to train five ML models, namely, the K-Nearest Neighbor (KNN), Support Vector Machine (SVM), Random Forest (RF), Neural Network (NN), and Convolutional Neural Network (CNN), to evaluate the best out of the ten defect identification models for identifying geometrically defective bead segments.

2.1. Experimental setup and data collection

Experiments were conducted on the robotic WAAM system at the Singapore University of Technology and Design (SUTD) to construct the geometric defect detection frameworks, as shown in Figure 3.

The system consists of a robot manipulator (ABB IRB1660ID), a welding power source (Fronius TPS 400i) equipped with a welding torch (Fronius WF 25i Robacta Drive), a Cartesian coordinate robot made up of three linear rails (PMI KM4510) powered by three servos (SmartMotor SM34165DT). In addition, a microphone (UMIK-1 miniDSP) is positioned at about 80 cm above the substrate for acoustic data measurement. The UMIK-1 acoustic microphone is an omnidirectional microphone that is pre-polarised. Its frequency response is 20 Hz–20 kHz and can capture sound signal sampling rates up to 48 kHz.
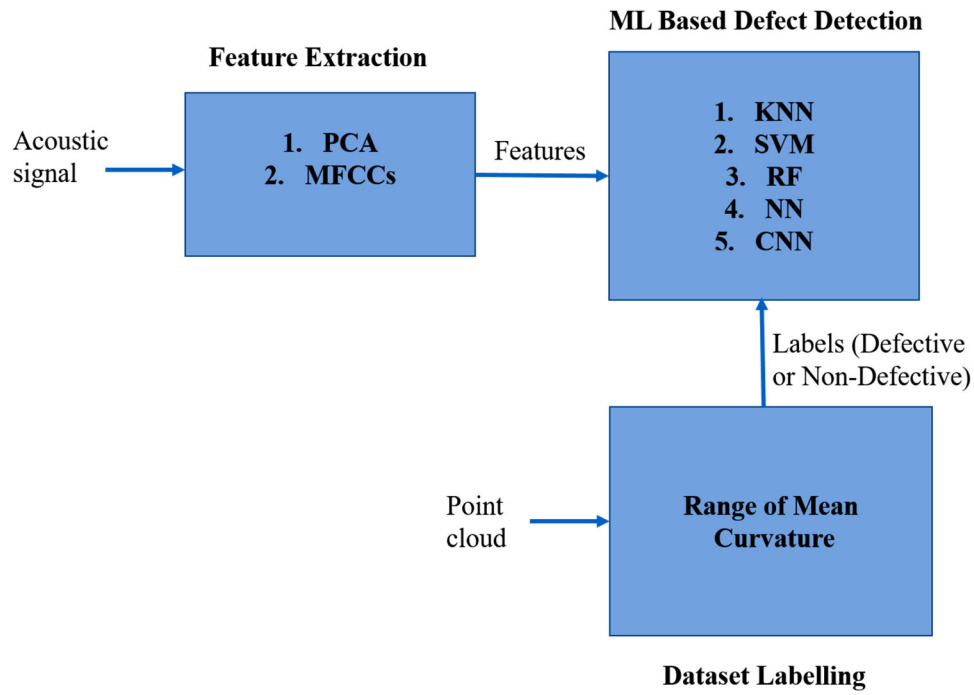
2.1.1. Data collection
Thirty-three weld beads of length 100 mm were printed using Inconel 718 (BÖHLER 3Dprint AM 718) wires. Each bead was printed using a different torch speed and wire feed rate combination to obtain different weld bead geometry. The torch speed (TS) and wire feed rate (WFR) were chosen based on the material process map that yielded good bead
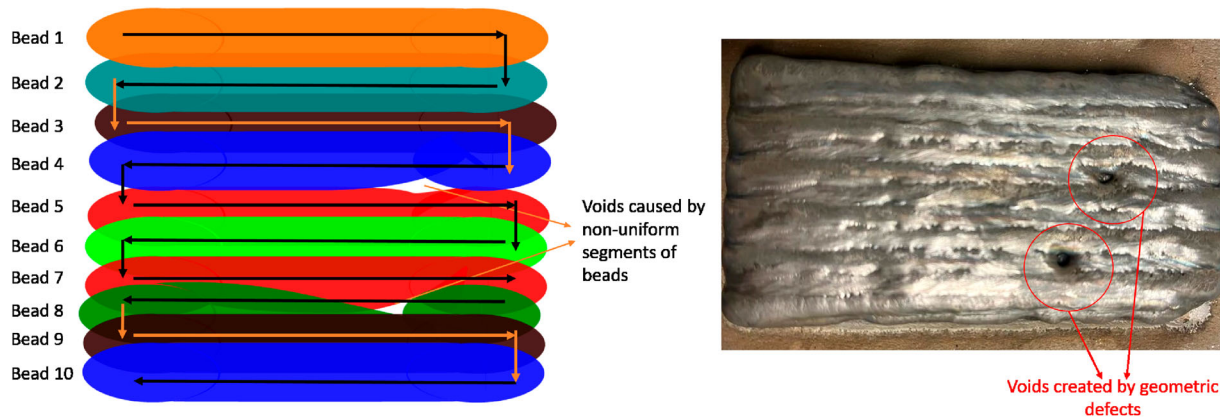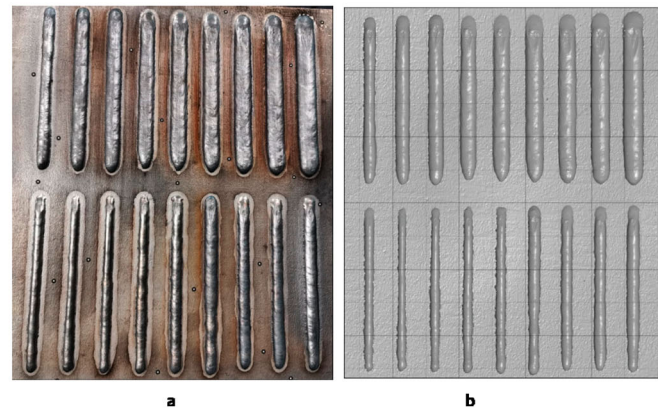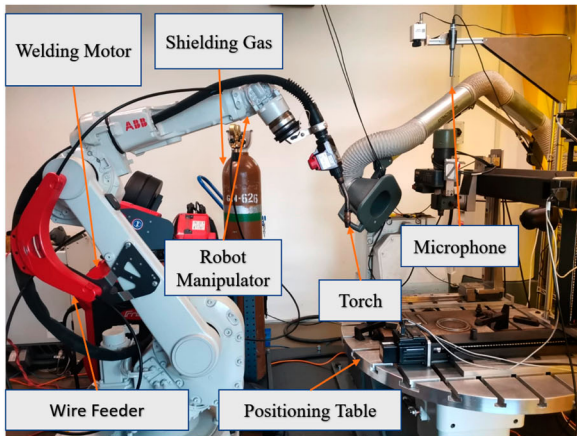

Figure 1. Schematic representation of a multi-beads printing (left) and Actual printing (right) as seen from the build direction. Geometrically defective segments lead to voids between two successive beads will affect the final printed part strength and quality.

Figure 2. The methodology for constructing the geometric defect detection Frameworks.

Figure 3. Experimental setup of SUTD Robotic WAAM for Bead Geometry and Acoustic Data Collection (left). (a) Original beads with plate (b) Mesh representation of beads with plate (right).

formation and was found to be in the range of [1,20] mm/s and [2,8] m/min, respectively. The shielding gas used a composition of 70% Ar and 30% He, and at a constant gas flow rate of ~ 25 L/min. The nozzle-to-work distance was set at around 15mm. For each bead print, acoustic signals were acquired at 44k Hz and divided into 20 signal segments. Since the different combinations of torch speed and wire feed rate produce different acoustic signals length, the temporal resolution of all beads is not the same. Among all the bead segments, the minimum temporal resolution is around 230ms and the maximum

temporal resolution is around 5 s. Therefore, downsampling (Surovi, Dharmawan, and Soh. 2021) was performed to make all the acoustic signal segments of equal length and at a sampling rate of 44100 Hz. Thus, a total of 660 acoustic signal segments for different torch speeds and wire feed rates were obtained.

Similarly, the GOM ATOS III Triple scanner was used for collecting point clouds measurement of the printed bead geometry to determine each bead segment curvature range. The point cloud data consists of both vertex and face information. Examples of the measured bead

point cloud are shown in Figure 3. Each bead was separated from its substrate using RANSEC plane segmentation and divided into 20 segments to yield 660 bead profile segments. The length of each bead segment is 5 mm. Together with the acoustic signal segments, the bead profile segments form a set of datasets for feature extraction, labeling, and construction of the defect identification model.

2.2. Acoustic feature extraction

The acoustic features consist of the Principle Components and Mel Frequency Cepstral Coefficients of the acoustic signal. In the following the mathematical background for each feature extraction technique and implementation details will be explained. Before extracting the features, the acoustic segments are denoised by wavelet threshold de-noising (Zhang, Wen, and Chen 2017).

2.2.1. Principle component analysis (PCA)
PCA is a popular statistical transform technique for multidimensional data. It is used extensively in the image and audio processing for reducing dimensionality and identifying signals of interest. PCA uses singular value decomposition and projects the high dimensional data into a lower dimension space (Tipping and Bishop 1999). The PCA feature extraction procedure consists of 5 steps and is given below:

- Step 1: Compute Discrete Fourier Transform (DFT) on the acoustic signal $x(t)$ using the following equation:

$$X(k) = \sum_{n=0}^{N-1} x(n)\, e^{-j2\pi nk/N}, \quad 0 \le k \le (N-1) \quad (1)$$

where N is the number of points used to compute the DFT.
- Step 2: Assemble the DFT signal of each segment y(k) into a matrix.

$$[Y] = \{y(0),\, y(2),\, \ldots,\, y(N-1)\} \quad (2)$$

- Step 3: Calculate the covariance matrix $[Q] = [Y][Y]^T$
- Step 4: Calculate the eigenvectors and eigenvalues of the covariance matrix using the following equation:

$$\lambda_i e_i = [Q]e_i \quad (3)$$

where λ_i is the eigenvalue associated with the eigenvector e_i.
- Sort the eigenvectors in the descending order of eigenvalues. The eigenvector with the highest eigenvalue forms the 1st principal component, the eigenvector with the second-highest eigenvalue forms the 2nd principal component, the eigenvector with the third-highest eigenvalue forms the 3nd principal component, and so on.

The dimension of the DFT features used for each bead segment is $N = 10495$. An example of the DFT feature of a signal segment based on this dimension is shown in Figure 4 (left). The number of principal components required as a feature depends on the ML model employed for geometric defect detection. In this paper, 4 principal components yields optimal results for KNN, SVM, NN and CNN models. As for RF, it is found that 6 principal components yield the best results.

2.2.2. Mel frequency cepstral coefficients (MFCC)
MFCC is a popular acoustic feature extraction technique and is used for speech and emotion recognition (Sato and Obuchi 2007). Mel frequency refers to human audible range frequency. MFCCs are short-term power spectrum-based features (Logan 2000) which capture the distinguishing characteristics of sound. The MFCCs feature extraction procedure (Rao and Manjunath 2017) from acoustic signal consists of 7 steps and is given below:

(1) Step 1: Convert each segmented signal into frames.
(2) Step 2: On each frame, apply a window function (e.g. Hamming window) to get a windowed signal.
(3) Step 3: Compute Discrete Fourier Transform (DFT) on the windowed signal using Equation (1).
(4) Step 4: Compute the Mel-filter bank, which is a set of band-pass filters. The filter bank is a non-linear-scale filter bank that imitates a human's audible system. Most of the filter shape is triangular. The filter banks are implemented in the frequency domain for MFCC computation. The conversion of physical frequency to Mel frequency is given by

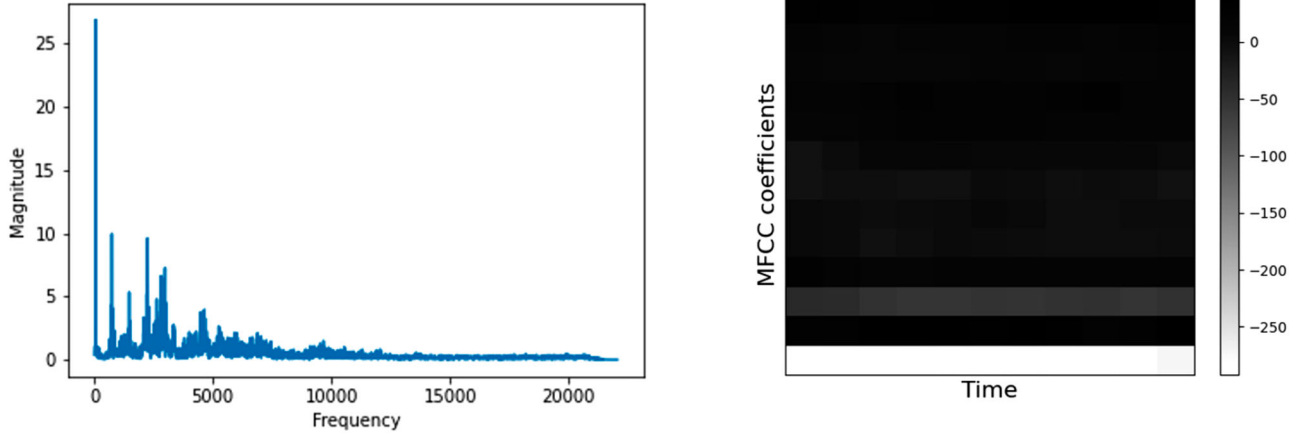$$f_{Mel} = 2595 \log_{10}(1 + f/700) \quad (4)$$

where f is the physical frequency in Hz and f_{Mel} is the human perceived frequency.
(5) Step 5: Calculate filter-bank energies by multiplying the square of the magnitude spectrum of the DFT signal with each filter-bank:

$$s(m) = \sum_{k=0}^{N-1} [X(k)]^2 H_m(k), \quad 0 \le m \le M-1 \quad (5)$$

where M is the total number of triangular Mel filters

Figure 4. DFT features of a segmented signal (left) MFCC features of a segmented signal (right).

and $H_m(k)$ is the weight given to the kth energy spectrum bin contributing to the mth output band.

(6) Step 6: Calculate the logarithm of the energies.

(7) Step 7: Apply Discrete Cosine Transform (DCT) to the log filter bank energies and produce the cepstral coefficients for each frame. Traditionally, 8 to 13 coefficients are selected. The equation for getting coefficients is given below:

$$c(n) = \sum_{m=0}^{M-1} \log_{10}(s(m)) \cos(\pi n(m-0.5)/M) \quad (6)$$

where $n = 0, 1, 2, \ldots, C-1$, c(n) are the cepstral coefficients, and C is the number of MFCCs.

The MFCC features can be computed using the *librosa* python package[1]. There are three parameters whose value needs to be set: frame size, hop length, and the number of filter banks, and those used in this paper are 2048, 512, and 13, respectively. The reason for setting the frame to 2048 is that if a frame size were smaller than this, the number of samples would not have been enough to get a reliable spectral estimate for defect detection. Similarly, if the frame size is larger than 2048, it may cause frequent changes in the information inside the frame, making the signal not stationary. As for the hop length, 512 is chosen to avoid any information loss at the beginning and end of the frame. Since the first few MFCC coefficients contain the majority of sound signal information compared to the rest (Gupta et al. 2013), 13 filter banks that give 13 MFCC coefficients are deemed sufficient for feature extraction. An example of the resulting MFCCs feature of a signal segment based on these parameters is shown in Figure 4 (right).

2.3. Dataset labeling based on range of mean curvature

The curvature at a point of a bead segment surface indicates how much it curves. The higher the curvature, the sharper the curve 'turns' at that point (Do Carmo 2016). The range of "mean curvature" of a segment, which we term range curvature, indicates how much the curvature changes in a bead segment. If the range curvature is small, the bead segment has a smooth or uniform surface. Otherwise, it indicates a non-uniform bead segment with some geometric defects. See Figure 5. Notice that the good bead segment has a smaller range curvature at 0.45 mm while the bad segment with a geometric defect (circled in red) has a higher range curvature at 1.2 mm.

To label the dataset, the range curvature of all bead segments needs to be computed and followed by a heuristic search to determine an optimal threshold to separate the good and bad segments. This can be achieved by searching through a set of range curvature threshold values. The optimal threshold value is such that the overlapping area formed by the Kernel Density Estimates (KDEs) of the good and bad bead segments yields minimal overlap. Based on this optimal threshold value, any bead segments above this threshold are labeled as bad. Otherwise, they are good.

2.3.1. Heuristic-based search for a range of mean curvature threshold

In the following, the details of the heuristic-based range curvature threshold search technique will be explained. Let $S = (s_1, s_2, \ldots, s_i, \ldots, s_n)$ be the sequence of bead segments, and $p_{i1}, p_{i2}, \ldots, p_{ij}, \ldots, p_{im}$ be the set of points on a particular bead segment s_i. The parametric spherical surface at each point on segment s_i is

Range of mean curvature=0.45 mm

Range of mean curvature=1.2 mm
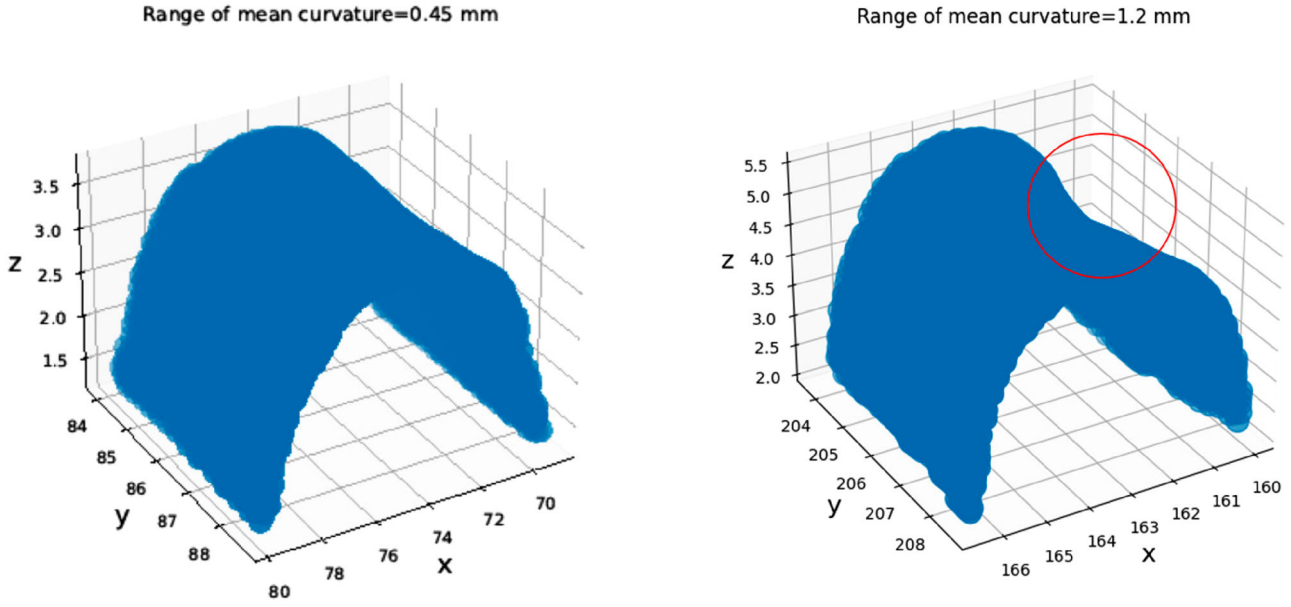
Figure 5. Range of mean curvature of geometrically good (left) and bad (right) bead segment. The red circle indicates a geometric defect.

approximated using the moving least squares (MLS) method (Guennebaud and Gross 2007).

To compute the mean curvature, κ, at each point of this parametric surface, let κ_{ij} be the mean-curvature at jth point p_{ij} on segment s_i. The mean curvature (Jia 2020) at this point, can be found as the average of the principal curvatures, κ_1 and κ_2.

$$\kappa_{ij}(p_{ij}) = \frac{1}{2}(\kappa_1(p_{ij}) + \kappa_2(p_{ij})) \qquad (7)$$

Such computation can be easily performed using MeshLab (Cignoni et al. 2008) with a colourise curvature (APSS) filter. The steps to determine the optimal range curvature threshold consist of 3 steps, which are given below. They consist of Range Curvature Parameter Metric Computation, Density Estimation, and Heuristic Search.

(1) Step 1: Let RC_i be the range of mean-curvatures of segment s_i. Compute range of mean-curvatures $(RC_1, RC_2, \ldots, RC_n)$ for n segments using

$$RC_i = \max_j(\kappa_{ij}) - \min_j(\kappa_{ij}) \quad i = 1, \ldots, n \qquad (8)$$

(2) Step 2: Let $G_i = \{s_j \in S : RC_j \leq RC_i\}$ be the set of good bead segments and $B_i = \{s_j \in S : RC_j > RC_i\}$ as the set of bad segments for range curvature RC_i. Estimate the distribution of feasibly good segments $\hat{f}(G_i)$ and bad segments $\hat{f}(B_i)$ using the

kernel density estimates function provided by *SciPy*[2] library in python.

(3) Step 3: To find the optimal RC_{i*}, heuristically search through various RC_i thresholds. The optimal RC_{i*} is the value that minimises the overlapping area between two KDE distributions of $\hat{f}(G_i)$ and $\hat{f}(B_i)$.

An example of the resulting labeled data is shown in Figure 6. Note that the identified range curvature threshold is 0.85 (Section 3.1) and bead segments 1–3 and 18–20 are deemed defective segments.

2.4. ML models for geometric defect identification

The proposed geometric defect identification models built on the five ML models: K-Nearest Neighbor (KNN), Support Vector Machine (SVM), Random Forest (RF), Neural Network (NN), and Convolutional Neural Network (CNN). In the following, the background for each model and implementation details will be explained.

2.4.1. K-Nearest neighbor (KNN)
KNN is a supervised learning technique. The algorithm selects the Kth nearest points of a new example by calculating the distance between all existing examples with the new example. The shortest distance from the
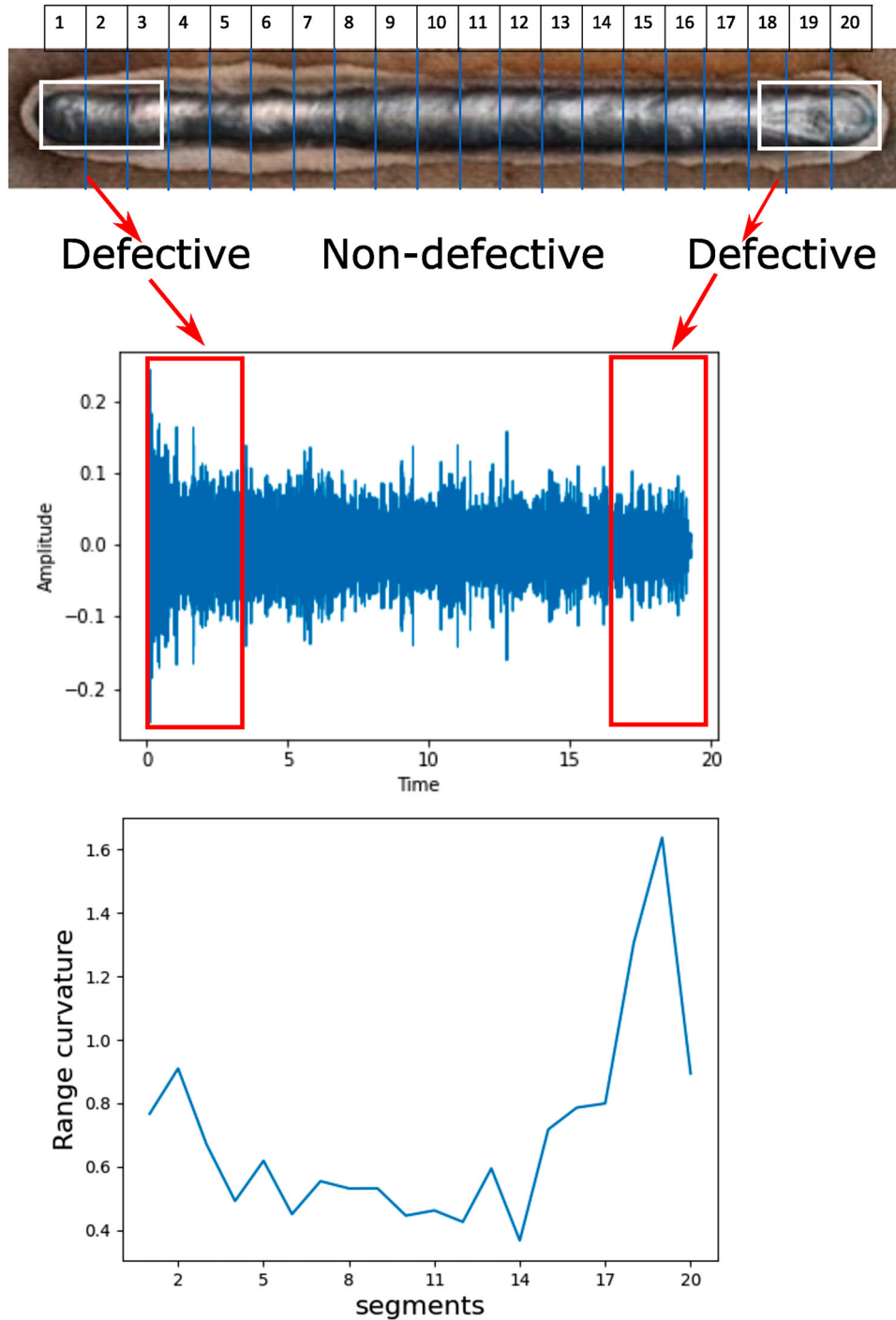
Figure 6. Bead with defective and non-defective segments (Top) with corresponding acoustic time-domain waveforms (Middle) and range curvature (Bottom).

new example determines the Kth nearest neighbours. The class of a new example is assigned to the majority class of Kth nearest points. For this application, the number of nearest neighbours is proposed to be $K = 11$ for PCA features and $K = 31$ for MFCC features. These are selected as they yield minimum mean validation loss over the range of K explored. See Section 3.2.1 for the experiments leading to such a choice.

2.4.2. Support vector machine (SVM)

SVM is a discriminative classifier that creates a line or a hyper-plane to separate data points into different classes. A non-linear SVM was employed to classify the bead segments into good and bad based on their input features and corresponding labels. The goal of SVM is to search a function $f(x)$ with parameters α_i and b (Platt 1999), which can predict defective segments of

beads. A general format of SVM is shown

$$f(x) = \sum_{i=1}^{n} y_i \alpha_i k(x_i, x) + b \qquad (9)$$

and the RBF kernel (radial basis function kernel) used is

$$k(x_i, x) = \exp\left(-\gamma \|\mathbf{x} - \mathbf{x_i}\|^2\right) \qquad (10)$$

For this application, the hyperparameter γ is selected to be 0.7 for PCA features, and 0.01 for MFCC features. See Section 3.2.2 for the experiments leading to such a choice.

2.4.3. Random forest (RF)

RF is a supervised ML technique with many decision trees as its building blocks. RF is suitable for modeling non-linear and complex systems. It is generally unaffected by outliers and noise and involves a faster training process. RF learns to build the relationships between the inputs and outputs in the training stage. Once the training is done, a new example is presented for testing. Each tree in the trained RF then votes for classification (Breiman 2001). The classification with the most votes is used as the predicted classification. For this application, the number of trees is selected as 150 for the PCA features and 60 for the MFCC features. See Section 3.2.3 for the experiments leading to such a choice.

2.4.4. Neural network (NN)

NN combines input layer neurons, hidden layer neurons, and output layer neurons, where two neurons are connected by weight. A set of input data is used to train the NN to optimise its desired output. The training process is done by adjusting the weights between the connected neurons to minimise the loss (Wang 2003).

The architecture of the proposed NN for defective segment identification comprises one input layer, four hidden layers and one output layer, as shown in Figure 7. Inputs to the NN are the extracted PCA features or MFCC features. A Rectified Linear Unit (ReLU) activation function (Nair and Hinton 2010) is used in all hidden layers. The L_2 regularisation is chosen as 0.001 and is used in every hidden layer to avoid the overfitting problem (Cortes, Mohri, and Rostamizadeh 2012). See

Section 3.2.4 for the experiments leading to such a framework.

2.4.5. Convolutional neural network (CNN)

CNN is analogous to traditional NN, comprised of convolutional, pooling, and fully connected layers. The main difference between CNN and traditional NN is that the input of CNN comprises images, and the neurons of CNN layers are three-dimensional-height, width, and depth information to the images (O'Shea and Nash 2015).
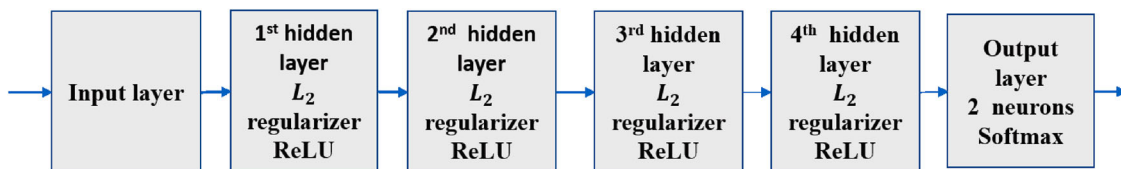
The architecture of the proposed CNN for defective segment identification comprises one convolutional layer, one pooling layer, one fully connected layer, and one output layer, as shown in Figure 8. Both the PCA and MFCC features are converted into input images. The convolutional layer consists of 13 and 5 filters for PCA and MFCC feature-based models, respectively. The size of each filter is 3×3. The Fully Connected (FC) layer contains 64 neurons to connect all the activation of previous layers. One dropout (0.3) is added after the FC layer to avoid overfitting. The final architecture is optimised with a suitable stride of 2×2 and regularisation (ReLU) of 0.001. One convolutional layer was sufficient for this application, as having two yields no significant accuracy improvement. See Section 3.2.5 for the experiments leading to such a framework.

3. Experimental results

In this section, experiments are conducted to determine the range curvature threshold for dataset labeling and the hyperparameters for the various proposed ML models for identifying geometrically defective bead segments during the WAAM process. Using the hyperparameters, a set of ML models is trained based on the identified feature parameters. A comparative study of all the trained models is subsequently evaluated for their defect identification performance.

3.1. Threshold selection for dataset labeling

The dataset labeling seeks to separate the dataset into good and bad bead segments based on their range

Figure 7. The proposed NN Architecture for defective bead segments identification.
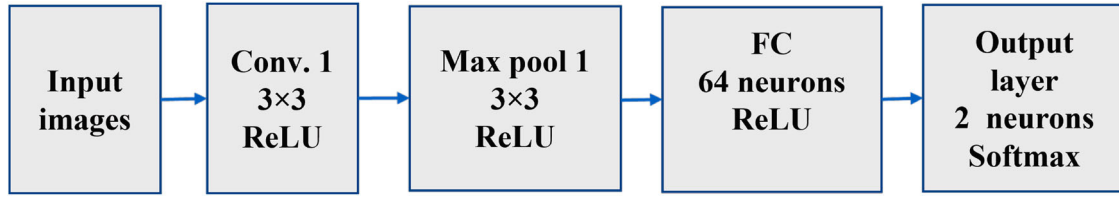
Figure 8. The proposed CNN Architecture for defective bead segments identification.

curvature. As explained previously, this can be achieved by searching through a set of range curvature threshold values that yield minimal overlapping area formed by the Kernel Density Estimates (KDEs) of the resulting good and bad bead segments. Figure 9 shows the KDE distribution plots for range curvature thresholds of 0.55, 0.85, and 0.95, and the relationship of different range curvature thresholds with their associated over-lapping areas. Notice that each of the KDE plots yields different overlapping areas. The overlapping area is minimal when the range curvature threshold is 0.85 mm. Thus, based on this threshold, any bead segments above this value are considered bad, and vice versa.

The dataset after the curvature-based labeling consists of 415 (62%) good bead segments and 245 (38%) bad bead segments.

3.2. Choice of hyperparameters

In the following, various experiments are conducted to determine the set of hyperparameters suitable for the KNN, SVM, RF, NN, and CNN defect detection model.

3.2.1. Hyperparameters for KNN
Figure 10 shows the effects of the KNN hyperparameters on the resulting classification error for both the PCA and
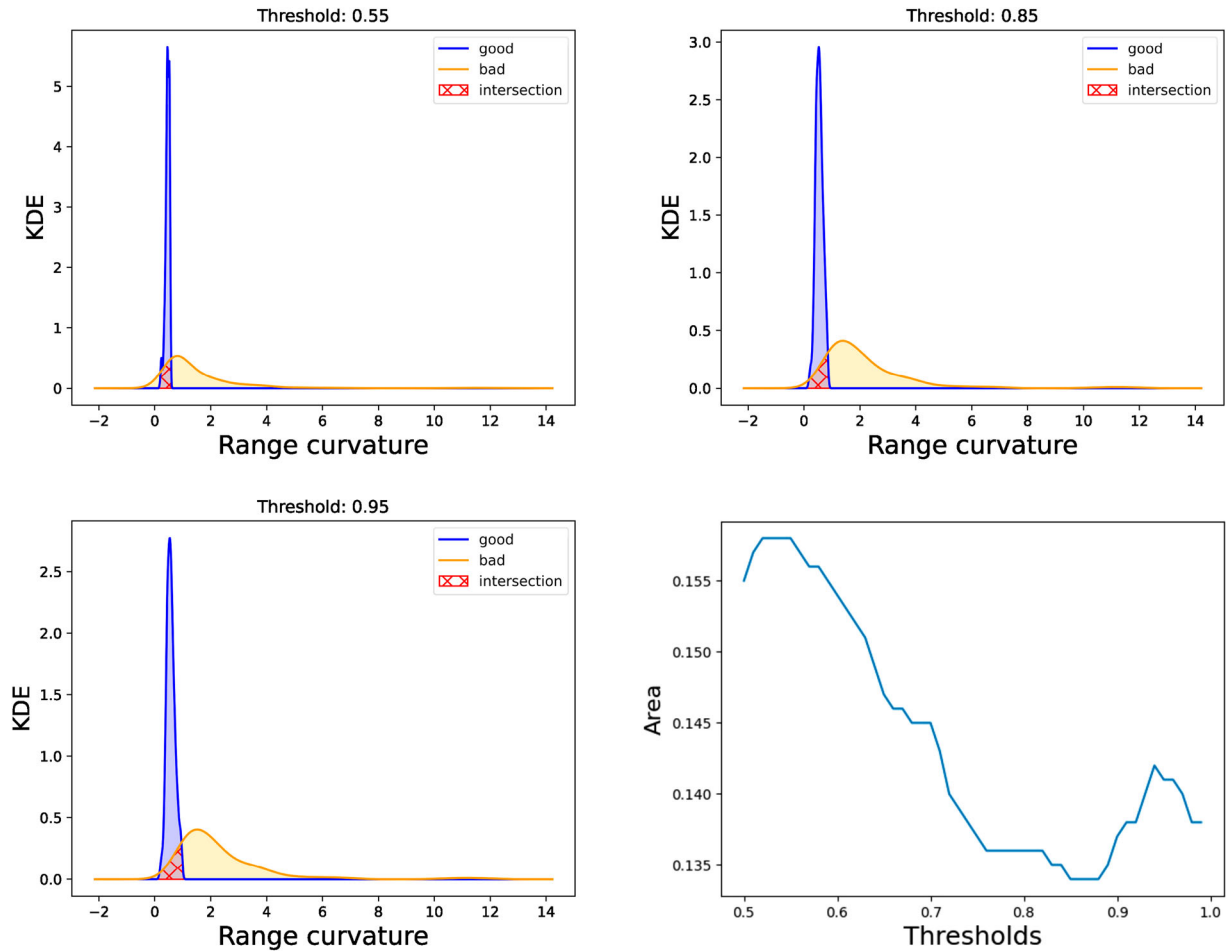
Figure 9. Distribution plots for good and bad segments of beads for different thresholds (First three plots). Different thresholds vs overlapping area plot(last plot).
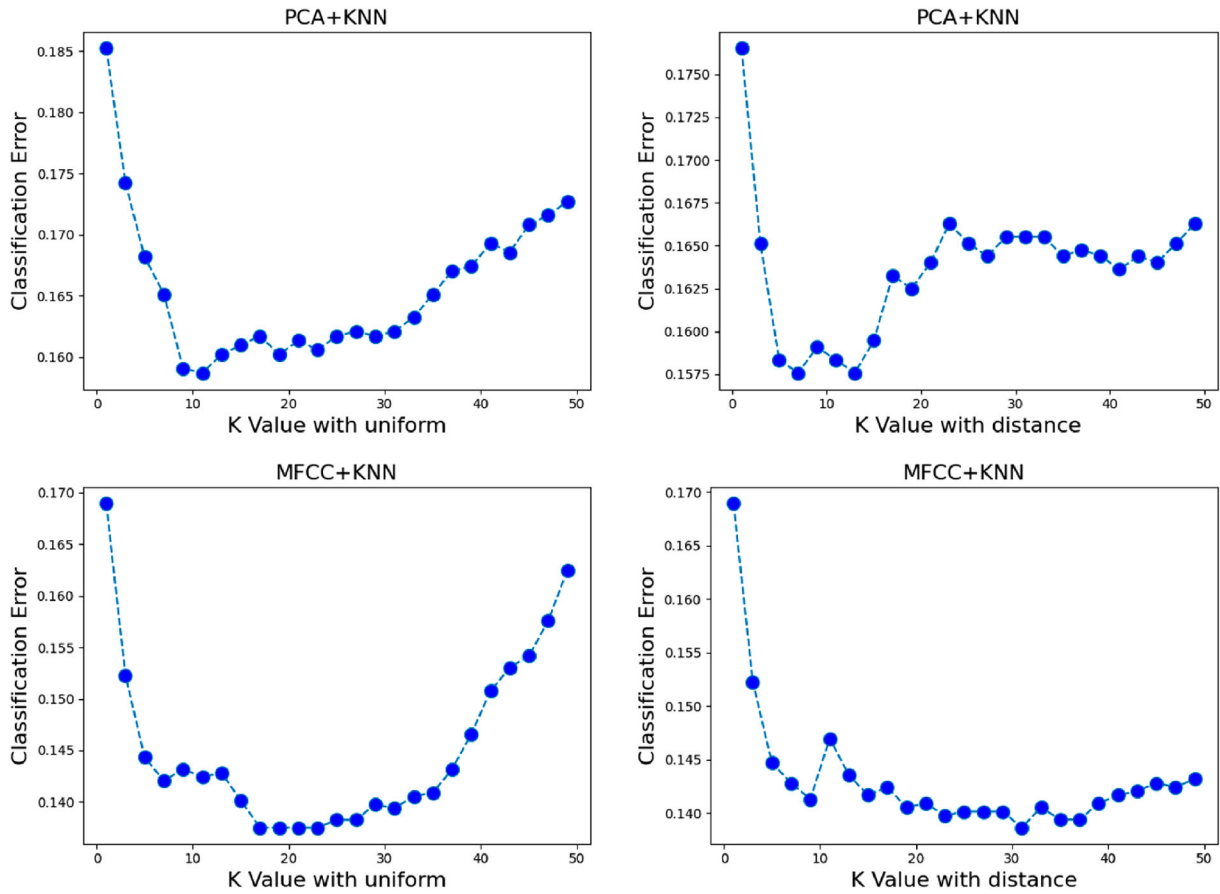
Figure 10. KNN classification error based on the number of nearest neighbour *K* and weight function for PCA features (top plot) and MFCC features (bottom plot).

MFCC features. The hyperparameters include the number of nearest neighbours *K* and the weight function. To find the optimal *K* and required weight function, experiments are conducted to determine the classification error over a range of *K* and weight function. Two weight functions, the 'uniform' or 'distance', are explored. Five-fold cross-validation is performed on the training dataset for $K = 1, \ldots, 50$ with the two weight functions, and the average validation classification error based on the PCA and MFCC features is plotted. The results show that the lowest classification error with PCA features is around 0.156 when $K = 11$ with a 'uniform' weight function. Similarly, the lowest classification error with MFCC features is around 0.138 when $K = 31$ with a "distance-weight function.
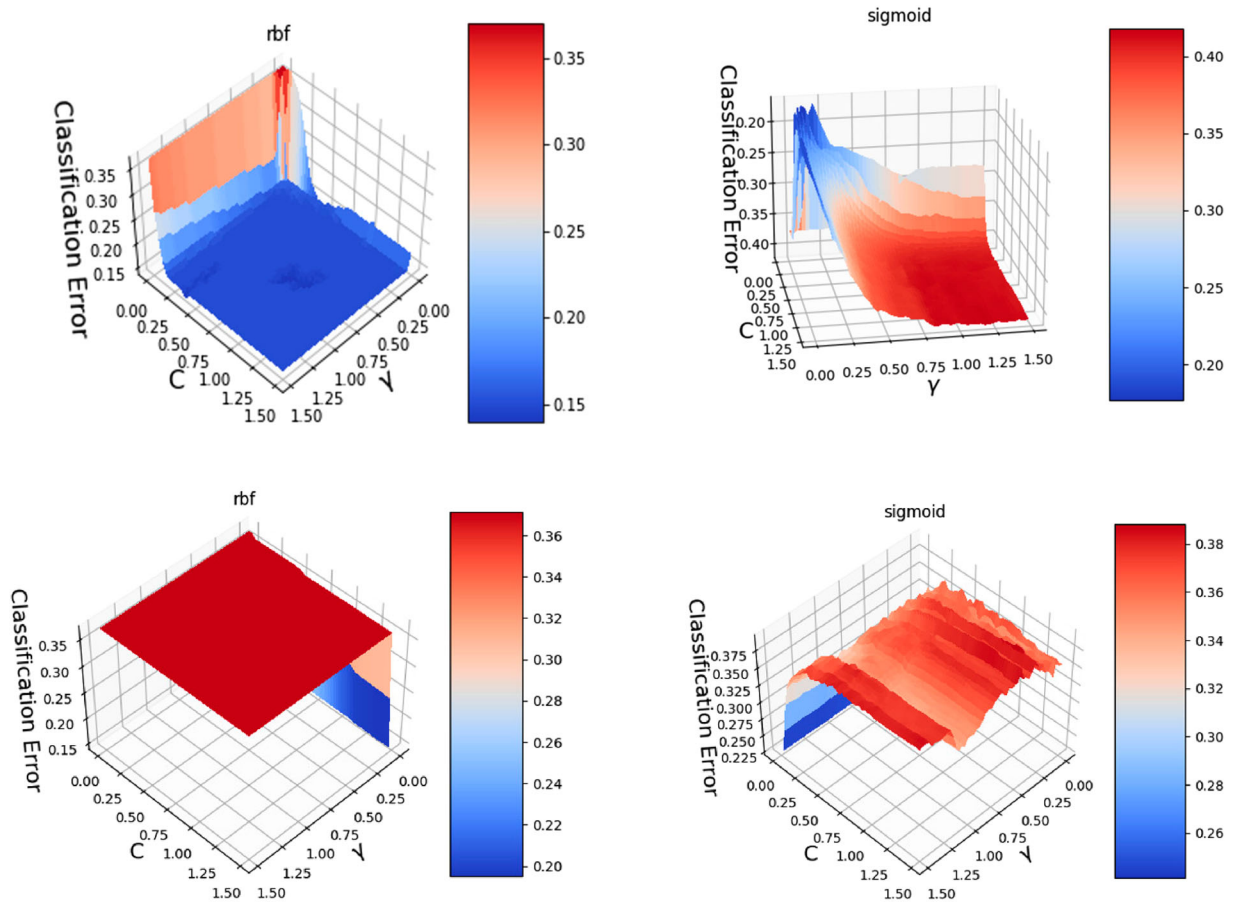
3.2.2. Hyperparameters for SVM
Figure 11 shows the effects of the SVM hyperparameters on the resulting classification error for both the PCA and MFCC features. The hyperparameters include a parameter *γ* that determines how far the influence of a single training example reaches, the regularisation parameter C, and the kernels. To find the optimal *γ*, C,

and required kernel function, experiments are conducted to determine the classification error over a range of *γ*, C, and kernels. Two kernels based on Radial Basis Function (RBF) and Sigmoid are explored. Five-fold cross-validation is performed on the training dataset for $\gamma = 0.01 \ldots 1.5$ and $C = 0.01 \ldots 1.5$ with the two kernel functions. The average validation classification error is plotted using PCA and MFCC features. The results show that the lowest classification error with a PCA feature occurs when $\gamma = 0.7$ and $C = 1.1$ with an RBF kernel. Similarly, the lowest classification error with an MFCC feature occurs when $\gamma = 0.01$ and C=1.48 with an RBF kernel.

3.2.3. Hyperparameters for RF
Figure 12 shows the effects of the RF hyperparameters on the resulting classification error for both the PCA and MFCC features. The hyperparameter consists of the number of trees used. Similarly, to find the optimal number of trees, five-fold cross-validation is performed on the training dataset as the number of trees varied from 1 to 350. The average validation classification error is plotted using PCA and MFCC features. The

Figure 11. SVM classification error based on the choice of *γ* and *C* for the RBF and Sigmoid kernels with PCA features (top) and MFCC features (bottom).

results show that the lowest classification error with a PCA feature is around 0.160 with 150 trees. Similarly, the lowest classification error with MFCC features is around 0.149 with 60 trees.

3.2.4. Hyperparameters of NN architecture

Figure 13, Tables 2, and 3 show the effects of the NN hyperparameters on the resulting classification accuracy

for both the PCA and MFCC features. The hyperparameters consist of the number of layers and neurons in each layer. Experiments are conducted to determine the validation accuracy as the number of layers and neurons varies to find the optimal number of layers and study the impact of the number of neurons in each layer. Five-fold cross-validation is performed on the training dataset as the number of layers varied

Figure 12. RF classification error based on the number of trees for PCA features (left plot) and MFCC features (right plot).
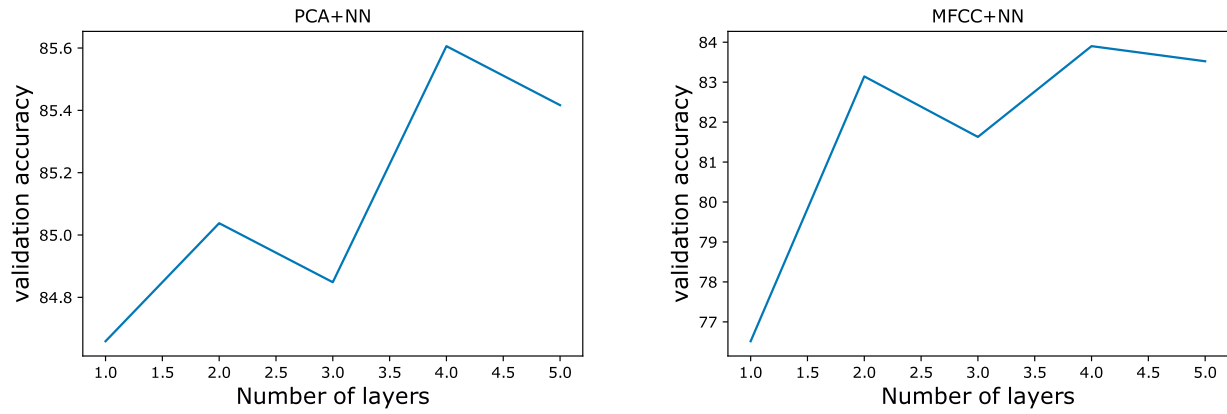
Figure 13. NN Classification accuracy based on the number of layers of a NN architecture for PCA features (left plot) and MFCC features (right plot).

from 1 to 5. The average validation classification accuracy is plotted based on PCA and MFCC features. The results show the 4-layer NN provides the highest classification accuracy based on the PCA and MFCC features.

Similarly, different combinations of neurons within a 4-layer NN are studied. In general, it is found that the validation accuracy is insensitive to neuron changes, with about 1% change based on the combinations tried. For PCA features, the 4-layer neurons combination of (256, 128, 64, 32) provides the highest classification accuracy. Similarly, for MFCC features, the 4-layer neurons combination of (256, 128, 64, 16) provides the highest classification accuracy.

3.2.5. Hyperparameters of CNN architecture

Figure 14 shows the effects of the CNN hyperparameters: 'the number of filters in the convolutional layer' on the resulting classification accuracy for both the PCA and MFCC features. To find the optimal number of filters, experiments are conducted to determine the classification accuracy as the number of filters varies. Five-fold cross-validation is performed on the training data set as the number of filters varied from 1 up to 40. The average validation classification error is plotted based
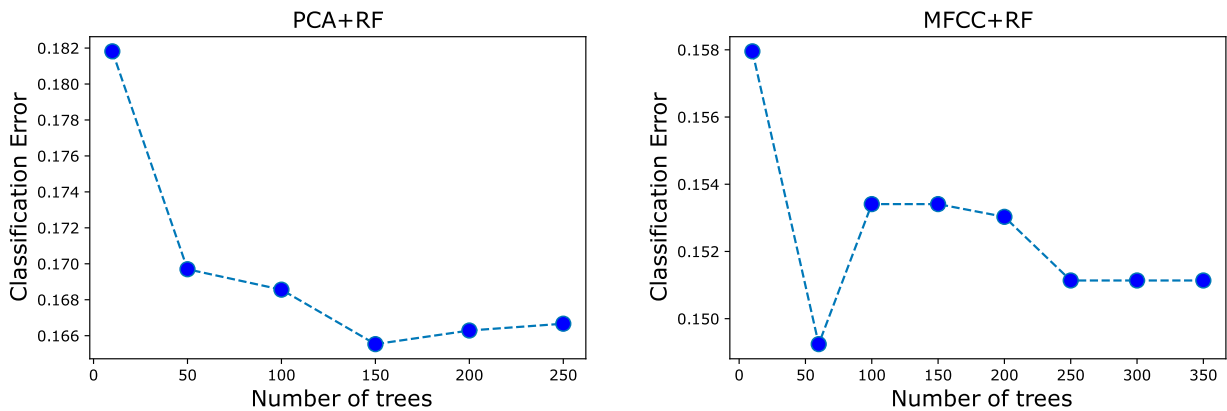
on PCA and MFCC features. The results show that the highest classification accuracy with a PCA feature occurs at 13 filters. Similarly, the highest classification accuracy with MFCC features occurs at 5 filters.

3.3. Training mechanism

With the set of hyperparameters for the various ML models determined in the earlier sections, a five-fold cross-validation (stratified k-fold) (Cawley and Talbot 2003) is used to train the defect detection models based on PCA and MFCC features, using the Inconel 718 bead segment dataset. For training on the NN and CNN model, adam optimiser (Kingma and Ba 2014) with sparse categorical cross-entropy loss, and the value of 0.0001 is used as the learning rate.

3.3.1. Defect identification models training based on PCA features

Figure 15 shows the average validation accuracy for each of the defect identification models as the principal components of the acoustic signal varied from 1 to 25.

Note that for the number of principal components taken as a feature, the average accuracy is based on

Table 2. Different Combination of number of neurons for PCA +NN.

layer 1	layer 2	layer 3	layer 4	validation accuracy
256	128	64	32	84.1
192	128	64	32	83.6
128	128	64	32	83.1
256	82	64	32	83.9
256	100	64	32	84.2
256	128	64	32	84.4
256	**128**	**64**	**32**	**84.5**
256	128	56	32	83.6
256	128	48	32	83.5
256	128	64	32	84.1
256	128	64	24	83.06
256	128	64	16	82.59

Table 3. Different Combination of number of neurons for MFCC +NN.

layer 1	layer 2	layer 3	layer 4	validation accuracy
256	**128**	**64**	**16**	**83.7**
192	128	64	16	83.5
128	128	64	16	83.2
256	82	64	16	83.4
256	100	64	16	83.6
256	128	64	16	82.8
256	128	64	16	83
256	128	56	16	82.4
256	128	48	16	82.1
256	128	64	32	81.8
256	128	64	24	82.2
256	128	64	16	82.9
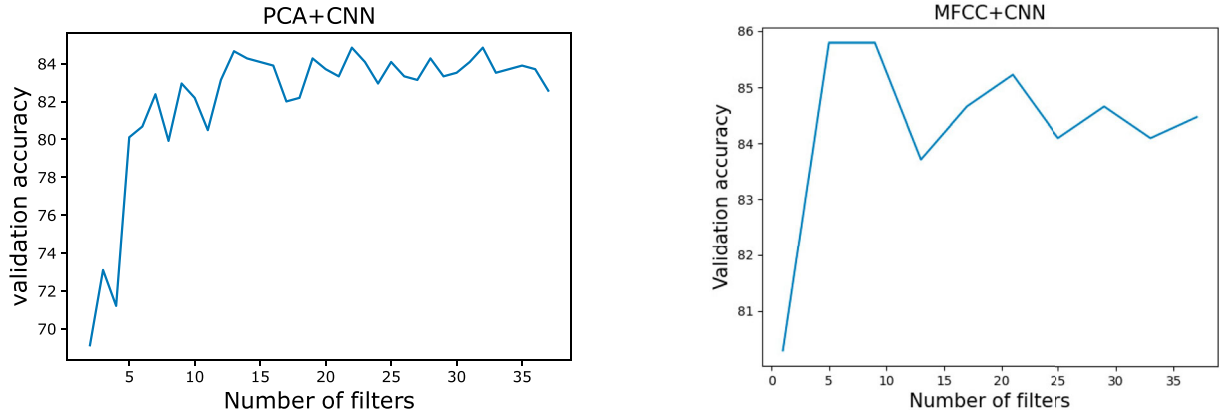
Figure 14. CNN classification accuracy based on the number of filters within its architecture for PCA features (left plot) and MFCC features (right plot).

the average of 30 randomly selected dataset of 60% training, 25% testing, and 15% validation. As shown from the results, the average validation accuracy increases at the beginning and reaches around 87% to 90% (depending on the type of ML model used). It shows that the first four components of PCA when combined as a feature, yield the most accurate prediction for KNN, SVM, NN, and CNN models. Similarly, the first six components of PCA when combined as a feature, yield the most accurate prediction for RF model.

Figure 16 shows the loss curve for NN and CNN models. Notice that both the training and testing loss decrease the number of epochs increase. It seems that after 35 epochs, both training and validation loss become stable with a small gap, indicating that the model had been trained without over-fitting. Thus, the training stops at 35 epochs.

3.3.2. Defect identification models training based on MFCCs features

Figure 16 shows the loss curve for the NN and CNN models. Notice that both training and testing loss decrease as the number of epochs increases. Similar to the PCA features, both training and testing loss become stable with a minimum gap after 35 epochs, and no overfitting are observed. Thus, the training stops at 35 epochs.

3.4. Performance evaluation of defect detection model based on F1 score, precision, recall and confusion matrix

To test the resulting trained defect detection model performance, the various models are evaluated based on
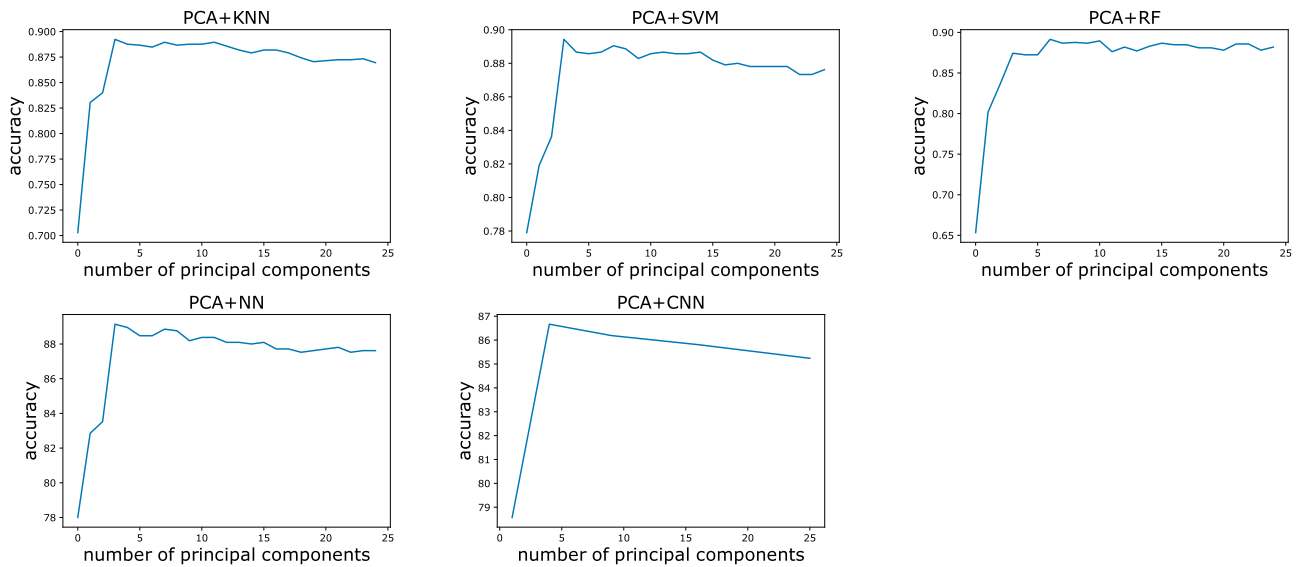
Figure 15. Average validation accuracy for the different defect identification models vs the number of principal components used as a feature.
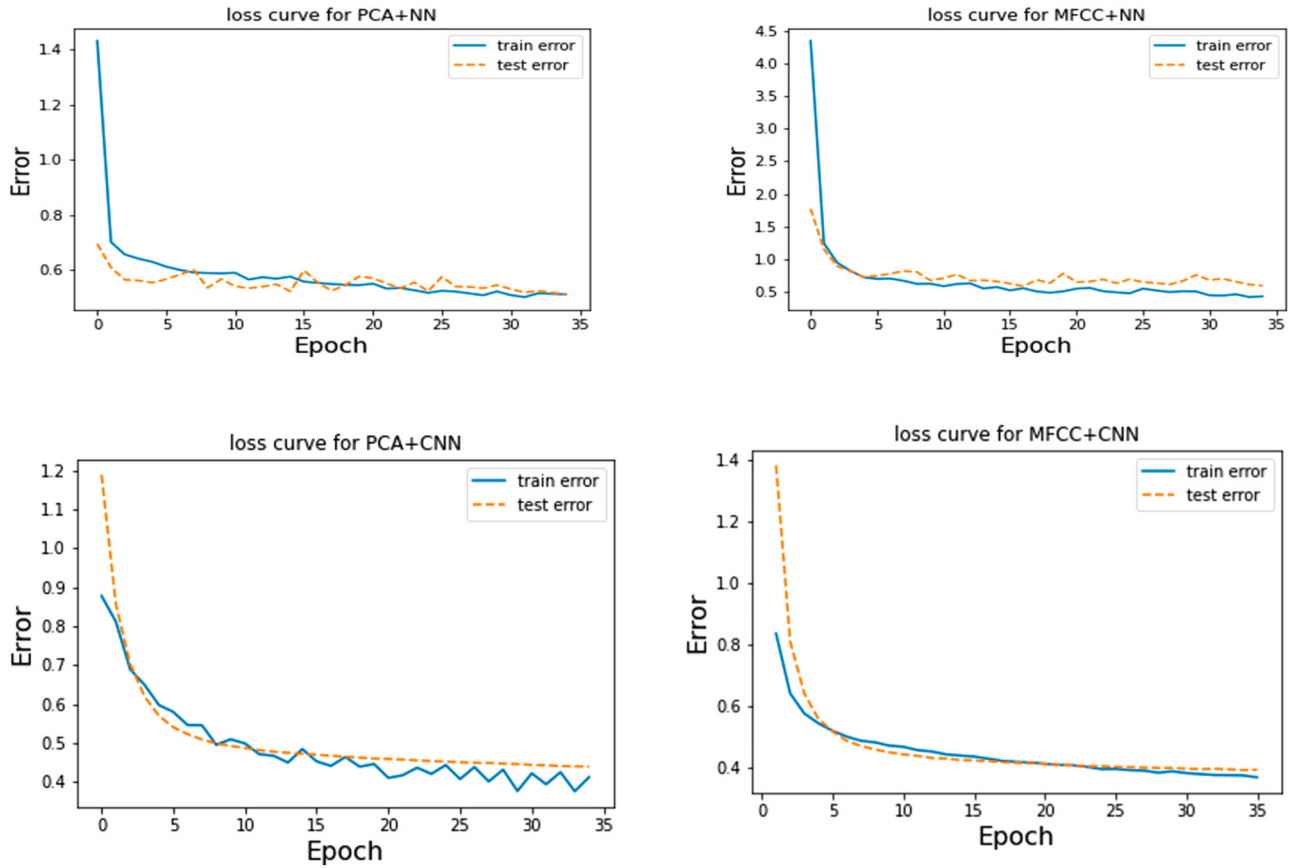
Figure 16. The loss curve for PCA+NN and MFCC+NN (top) The loss curve for PCA+CNN and MFCC+CNN (bottom).

their F1 score, precision, recall, and confusion matrix. The reason for using these metrics is that they are useful in evaluating the performance of an imbalanced dataset. Table 4 shows the comparative study performed on the resulting defect identification models based on F1 score, precision, and recall.

It is observed that the MFCC feature with an SVM model (MFCC+SVM) has the highest F1 score among all the defect identification models, at around 84%.

Table 5 shows the comparative study performed on the resulting defect detection models based on the

confusion matrix, which measures the truly good and bad segments picked out by the models. It shows that the PCA feature with a KNN model (PCA+KNN) can pick out the most number of true bad bead segments at 186.

Table 5. Comparative study of all of the proposed defect identification model based on Confusion Matrix.

Defect Identification Model		True		
			Good	Bad
PCA+KNN	Predicted	Good	354	59
		Bad	61	186
PCA+SVM	Predicted	Good	400	83
		Bad	15	162
PCA+RF	Predicted	Good	366	68
		Bad	49	177
PCA+NN	Predicted	Good	381	68
		Bad	34	177
PCA+CNN	Predicted	Good	388	77
		Bad	27	168
MFCC+KNN	Predicted	Good	393	75
		Bad	22	170
MFCC+SVM	Predicted	Good	405	80
		Bad	10	165
MFCC+RF	Predicted	Good	395	73
		Bad	20	172
MFCC+NN	Predicted	Good	369	76
		Bad	46	169
MFCC+CNN	Predicted	Good	399	77
		Bad	16	168

Table 4. Comparative study of all of the proposed defect identification model based on F1 score, precision, and recall.

Defect Identification Model	F1 Score	Precision	Recall
PCA+KNN	80.59 ± 0.05	81.86 ± 0.03	80.06 ± 0.11
PCA+SVM	83.50 ± 0.08	87.06 ± 0.08	81.29 ± 0.06
PCA+RF	80.71 ± 0.04	81.18 ± 0.03	80.21 ± 0.11
PCA+NN	82.80 ± 0.03	84.41 ± 0.08	82.11 ± 0.05
PCA+CNN	82.26 ± 0.06	84.63 ± 0.06	80.01 ± 0.05
MFCC+KNN	83.19 ± 0.08	87.15 ± 0.06	81.62 ± 0.07
MFCC+SVM	84.24 ± 0.08	88.65 ± 0.06	82.65 ± 0.01
MFCC+RF	84.18 ± 0.03	87.28 ± 0.03	82.10 ± 0.07
MFCC+NN	81.21 ± 0.17	82.44 ± 0.05	80.67 ± 0.02
MFCC+CNN	79.45 ± 0.07	82.55 ± 0.08	77.44 ± 0.11

4. Conclusion

In this paper, a methodology for constructing the geometric defect detection model to identify defective bead segments using acoustic sensing for the WAAM process is proposed. The performance of ten of such models is evaluated, and the results have demonstrated they can able to identify geometrically defective segments accurately (80% to 85%) when tested on the Inconel 718 dataset. Among the various trained defect detection models, it is found that having MFCC acoustic features with support vector machine performs the best in terms of F1 score, and having principal components as an acoustic feature with K nearest neighbour performs the best based on the confusion matrix. The first novelty of this research lies in the defects that are targeted. Geometric defects are a unique problem in WAAM due to their sensitivity to process variation. A bead produced using a constant process parameter would produce bead segments of different geometrical shapes due to environmental factors. This would lead to voids and affect the final part performance if left untreated. The second novelty lies in the segment-wise geometric defect detection approach. By discretizing defect detection, it has the benefit of identifying a localised defect more accurately so that early intervention can be possible. The third novelty lies in the dataset labeling approach of geometric defects, where good and bad bead segments are separated based on a threshold of the range of mean curvature determined through a heuristic search. Currently, labeling for AM defects is based on a visual inspection approach, which is error-prone and time-consuming when the dataset is large. Here an approach to determine an optimal threshold for labeling based on overlapping areas of KDE distribution is proposed to separate good and bad segments. For future work, the authors intend to explore the inclusion of other features, such as the WAAM process parameters, to further increase the defect detection performance. The goal is to implement this approach for defect detection during real-life printing.

Notes

1. https://librosa.org/
2. https://scipy.org//

Disclosure statement

No potential conflict of interest was reported by the author(s).

ORCID

Nowrin Akter Surovi http://orcid.org/0000-0002-8481-9923
Gim Song Soh http://orcid.org/0000-0002-0042-5151

References

Bevans, Benjamin, André Ramalho, Ziyad Smoqi, Aniruddha Gaikwad, Telmo G. Santos, Prahalad Rao, and J. P. Oliveira. 2023. "Monitoring and Flaw Detection During Wire-based Directed Energy Deposition Using in-situ Acoustic Sensing and Wavelet Graph Signal Analysis." *Materials & Design* 225: Article ID 111480. doi:10.1016/j.matdes.2022.111480.

Breiman, Leo. 2001. "Random Forests." *Machine Learning* 45 (1): 5–32. doi:10.1023/A:1010933404324.

Busachi, Alessandro, John Erkoyuncu, Paul Colegrove, Filomeno Martina, and Jialuo Ding. 2015. "Designing a WAAM Based Manufacturing System for Defence Applications." *Procedia Cirp* 37: 48–53. doi:10.1016/j.procir.2015.08.085.

Cawley, Gavin C., and Nicola L. C. Talbot. 2003. "Efficient Leave-One-Out Cross-Validation of Kernel Fisher Discriminant Classifiers." *Pattern Recognition* 36 (11): 2585–2592. doi:10.1016/S0031-3203(03)00136-5.

Chen, Lequn, Xiling Yao, Chaolin Tan, Weiyang He, Jinlong Su, Fei Weng, Youxiang Chew, Nicholas Poh Huat Ng, and Seung Ki Moon. 2023. "In-Situ Crack and Keyhole Pore Detection in Laser Directed Energy Deposition Through Acoustic Signal and Deep Learning." *Additive Manufacturing*, Article ID 103547. doi:10.1016/j.addma.2023.103547.

Chen, Lequn, Xiling Yao, Peng Xu, Seung Ki Moon, and Guijun Bi. 2021. "Rapid Surface Defect Identification for Additive Manufacturing With In-Situ Point Cloud Processing and Machine Learning." *Virtual and Physical Prototyping* 16 (1): 50–67. doi:10.1080/17452759.2020.1832695.

Cho, Hae-Won, Seung-Jun Shin, Gi-Jeong Seo, Duck Bong Kim, and Dong-Hee Lee. 2022. "Real-Time Anomaly Detection Using Convolutional Neural Network in Wire Arc Additive Manufacturing: Molybdenum Material." *Journal of Materials Processing Technology* 302: Article ID 117495. doi:10.1016/j.jmatprotec.2022.117495.

Chu, Hui-Hui, and Zong-Yi Wang. 2016. "A Vision-Based System for Post-Welding Quality Measurement and Defect Detection." *The International Journal of Advanced Manufacturing Technology* 86 (9): 3007–3014. doi:10.1007/s00170-015-8334-1.

Cignoni, Paolo, Marco Callieri, Massimiliano Corsini, Matteo Dellepiane, Fabio Ganovelli, and Guido Ranzuglia. 2008. "MeshLab: An Open-Source Mesh Processing Tool." In *Eurographics Italian Chapter Conference*, edited by Vittorio

Scarano, Rosario De Chiara, and Ugo Erra. The Eurographics Association.

Cortes, Corinna, Mehryar Mohri, and Afshin Rostamizadeh. 2012. "L2 Regularization for Learning Kernels." *arXiv preprint arXiv:1205.2653*.

Ding, Donghong, Zengxi Pan, Dominic Cuiuri, and Huijun Li. 2015. "A Practical Path Planning Methodology for Wire and Arc Additive Manufacturing of Thin-walled Structures." *Robotics and Computer-Integrated Manufacturing* 34: 8–19. doi:10.1016/j.rcim.2015.01.003.

Do Carmo, Manfredo P. 2016. *Differential Geometry of Curves and Surfaces: Revised and Updated Second Edition*. New Jersey: Prentice Hall.

Drissi-Daoudi, Rita, Giulio Masinelli, Charlotte de Formanoir, Kilian Wasmer, Jamasp Jhabvala, and Roland E. Logé. 2023. "Acoustic Emission for the Prediction of Processing Regimes in Laser Powder Bed Fusion, and the Generation of Processing Maps." *Additive Manufacturing* 67: Article ID 103484. doi:10.1016/j.addma.2023.103484.

Drissi-Daoudi, Rita, Vigneashwara Pandiyan, Roland Logé, Sergey Shevchik, Giulio Masinelli, Hossein Ghasemi-Tabasi, Annapaola Parrilli, and Kilian Wasmer. 2022. "Differentiation of Materials and Laser Powder Bed Fusion Processing Regimes From Airborne Acoustic Emission Combined with Machine Learning." *Virtual and Physical Prototyping* 17 (2): 181–204. doi:10.1080/17452759.2022. 2028380.

Gaja, Haythem, and Frank Liou. 2017. "Defects Monitoring of Laser Metal Deposition Using Acoustic Emission Sensor." *The International Journal of Advanced Manufacturing Technology* 90 (1–4): 561–574. doi:10.1007/s00170-016-9366-x.

Goh, Guo Dong, Swee Leong Sing, and Wai Yee Yeong. 2021. "A Review on Machine Learning in 3D Printing: Applications, Potential, and Challenges." *Artificial Intelligence Review* 54 (1): 63–94. doi:10.1007/s10462-020-09876-9.

Grad, Ladislav, Janez Grum, Ivan Polajnar, and Janez Marko Slabe. 2004. "Feasibility Study of Acoustic Signals for on-line Monitoring in Short Circuit Gas Metal Arc Welding." *International Journal of Machine Tools and Manufacture* 44 (5): 555–561. doi:10.1016/j.ijmachtools. 2003.10.016.

Guennebaud, Gaël, and Markus Gross. 2007. "Algebraic Point Set Surfaces." In *ACM SIGGRAPH 2007 papers*, 23–es.

Gupta, Shikha, Jafreezal Jaafar, W. F. Wan Ahmad, and Arpit Bansal. 2013. "Feature Extraction Using MFCC." *Signal & Image Processing: An International Journal* 4 (4): 101–108. doi:10.5121/sipij.2013.4408.

Huang, Cheng, Guilan Wang, Hao Song, Runsheng Li, and Haiou Zhang. 2022. "Rapid Surface Defects Detection in Wire and Arc Additive Manufacturing Based on Laser Profilometer." *Measurement* 189: Article ID 110503. doi:10. 1016/j.measurement.2021.110503.

Ito, Kaita, Masahiro Kusano, Masahiko Demura, and Makoto Watanabe. 2021. "Detection and Location of Microdefects During Selective Laser Melting by Wireless Acoustic Emission Measurement." *Additive Manufacturing* 40: Article ID 101915. doi:10.1016/j.addma.2021.101915.

Jia, Yan-Bin. 2020. "Gaussian and Mean Curvatures." *Com S* 477 (577): 1–7.

Kaushik, Balakrishnan, Don Nance, and Krish Ahuja. 2005. "A Review of the Role of Acoustic Sensors in the Modern Battlefield." In *11th AIAA/CEAS Aeroacoustics Conference*, 2997, Monterey, California.

Kingma, Diederik P., and Jimmy Ba. 2014. "Adam: A Method for Stochastic Optimization." *arXiv preprint arXiv:1412.6980*.

Kononenko, Denys Y, Viktoriia Nikonova, Mikhail Seleznev, Jeroen van den Brink, and Dmitry Chernyavsky. 2023. "An in Situ Crack Detection Approach in Additive Manufacturing Based on Acoustic Emission and Machine Learning." *Additive Manufacturing Letters* 5: Article ID 100130. doi:10.1016/j.addlet.2023.100130.

Kryukov, Igor, Michael Hartmann, Stefan Böhm, Malte Mund, Klaus Dilger, and Fabian Fischer. 2014. "Defect Detection in Friction Stir Welding by Online Infrared Thermography." *Journal of Welding and Joining* 32 (5): 480–487. doi:10. 5781/JWJ.2014.32.5.50.

Li, Wenhao, Haiou Zhang, Guilan Wang, Gang Xiong, Meihua Zhao, Guokuan Li, and Runsheng Li. 2023. "Deep Learning Based Online Metallic Surface Defect Detection Method for Wire and Arc Additive Manufacturing." *Robotics and Computer-Integrated Manufacturing* 80: Article ID 102470. doi:10.1016/j.rcim.2022.102470.

Logan, Beth. 2000. "Mel Frequency Cepstral Coefficients for Music Modeling." In *Ismir*, Vol. 270, 1–11. Citeseer.

Lv, Na, Jiyong Zhong, Huabin Chen, Tao Lin, and Shanben Chen. 2014. "Real-time Control of Welding Penetration During Robotic GTAW Dynamical Process by Audio Sensing of Arc Length." *The International Journal of Advanced Manufacturing Technology* 74 (1–4): 235–249. doi:10.1007/s00170-014-5875-7.

Nair, Vinod, and Geoffrey E. Hinton. 2010. "Rectified Linear Units Improve Restricted Boltzmann Machines." In *Proceedings of the 27th international conference on machine learning (ICML-10)*. Haifa, Israel on June 21-24, 807–814.

O'Shea, Keiron, and Ryan Nash. 2015. "An Introduction to Convolutional Neural Networks." *arXiv preprint arXiv:1511.08458*.

Pal, Kamal, Sandip Bhattacharya, and Surjya K. Pal. 2010. "Investigation on Arc Sound and Metal Transfer Modes for on-line Monitoring in Pulsed Gas Metal Arc Welding." *Journal of Materials Processing Technology* 210 (10): 1397–1410. doi:10.1016/j.jmatprotec.2010.03.029.

Palaz, Dimitri, and Ronan Collobert. 2015. *Analysis of Cnn-Based Speech Recognition System Using Raw Speech as Input*. Technical Report. Idiap.

Pandiyan, Vigneashwara, Rita Drissi-Daoudi, Sergey Shevchik, Giulio Masinelli, Tri Le-Quang, Roland Logé, and Kilian Wasmer. 2021. "Semi-supervised Monitoring of Laser Powder Bed Fusion Process Based on Acoustic Emissions." *Virtual and Physical Prototyping* 16 (4): 481–497. doi:10. 1080/17452759.2021.1966166.

Platt, John. 1999. "Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods." *Advances in Large Margin Classifiers* 10 (3): 61–74.

Polajnar, Ivan, Zoran Bergant, and Janez Grum. 2013. "Arc Welding Process Monitoring by Audible Sound." In *12th International Conference of the Slovenian Society for Non-Destructive Testing: Application of Contemporary Non-Destructive Testing in Engineering, ICNDT 2013-Conference*

Proceedings, Held 4-6 September 2013, Portoroz, Slovenia, 613–620.

Rajashekar, R., and B. M. Rajaprakash. 2016. "Development of a Model for Friction Stir Weld Quality Assessment Using Machine Vision and Acoustic Emission Techniques." *Journal of Materials Processing Technology* 229: 265–274. doi:10.1016/j.jmatprotec.2015.09.030.

Rao, Jing, Swee Leong Sing, Joel Choon We Lim, Wai Yee Yeong, Jizhong Yang, Zheng Fan, and Paul Hazell. 2022. "Detection and Characterisation of Defects in Directed Energy Deposited Multi-material Components Using Full Waveform Inversion and Reverse Time Migration." *Virtual and Physical Prototyping* 17 (4): 1047–1057. doi:10.1080/17452759.2022.2086142.

Rao, K. Sreenivasa, and K. E. Manjunath. 2017. *Speech Recognition Using Articulatory and Excitation Source Features.* Springer.

Rodrigues, Tiago A., V. Duarte, R. M. Miranda, Telmo G. Santos, and J. P. Oliveira. 2019. "Current Status and Perspectives on Wire and Arc Additive Manufacturing (WAAM)." *Materials* 12 (7): 1121. doi:10.3390/ma12071121.

Sato, Nobuo, and Yasunari Obuchi. 2007. "Emotion Recognition Using Mel-frequency Cepstral Coefficients." *Information and Media Technologies* 2 (3): 835–848. doi:10.5715/jnlp.14.4_83.

Shevchik, Sergey A, Giulio Masinelli, Christoph Kenel, Christian Leinenbach, and Kilian Wasmer. 2019. "Deep Learning for in Situ and Real-time Quality Monitoring in Additive Manufacturing Using Acoustic Emission." *IEEE Transactions on Industrial Informatics* 15 (9): 5194–5203. doi:10.1109/TII.9424.

Surovi, Nowrin Akter, Audelia G. Dharmawan, and Gim Song Soh. 2021. "A Study on the Acoustic Signal Based Frameworks for the Real-Time Identification of Geometrically Defective Wire Arc Bead." In *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Vol. 85383, V03AT03A003. American Society of Mechanical Engineers.

Surovi, Nowrin Akter, Shaista Hussain, and Gim Song Soh. 2022. "A Study of Machine Learning Framework for Enabling Early Defect Detection in Wire Arc Additive Manufacturing Processes." In *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Vol. 86229, V03AT03A002. American Society of Mechanical Engineers.

Surovi, Nowrin Akter, and Gim Song Soh. 2022. "Process Map Generation of Geometrically Uniform Beads Using Support Vector Machine." *Materials Today: Proceedings* 70: 113–118. doi:10.1016/j.matpr.2022.08.557.

Tempelman, Joshua R., Adam J. Wachtor, Eric B. Flynn, Phillip J. Depond, Jean-Baptiste Forien, Gabe M. Guss, Nicholas P. Calta, and Manyalibo J. Matthews. 2022. "Detection of Keyhole Pore Formations in Laser Powder-bed Fusion Using Acoustic Process Monitoring Measurements." *Additive Manufacturing* 55: Article ID 102735. doi:10.1016/j.addma.2022.102735.

Tipping, Michael E., and Christopher M. Bishop. 1999. "Probabilistic Principal Component Analysis." *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 61 (3): 611–622. doi:10.1111/1467-9868.00196.

Vilček, I., J. Řehoř, D. Carou, and P. Zeman. 2017. "Residual Stresses Evaluation in Precision Milling of Hardened Steel Based on the Deflection-electrochemical Etching Technique." *Robotics and Computer-Integrated Manufacturing* 47: 112–116. doi:10.1016/j.rcim.2016.10.001.

Wang, Sun-Chong. 2003. "Artificial Neural Network." In *Interdisciplinary Computing in Java Programming*, 81–100. Springer.

Wasmer, Kilian, Matthias Wüst, Di Cui, Giulio Masinelli, Vigneashwara Pandiyan, and Sergey Shevchik. 2023. "Monitoring of Functionally Graded Material During Laser Directed Energy Deposition by Acoustic Emission and Optical Emission Spectroscopy Using Artificial Intelligence." *Virtual and Physical Prototyping* 18 (1): e2189599. doi:10.1080/17452759.2023.2189599.

Wu, Bintao, Zengxi Pan, Donghong Ding, Dominic Cuiuri, Huijun Li, Jing Xu, and John Norrish. 2018. "A Review of the Wire Arc Additive Manufacturing of Metals: Properties, Defects and Quality Improvement." *Journal of Manufacturing Processes* 35: 127–139. doi:10.1016/j.jmapro.2018.08.001.

Xia, Chunyang, Zengxi Pan, Joseph Polden, Huijun Li, Yanling Xu, Shanben Chen, and Yuming Zhang. 2020. "A Review on Wire Arc Additive Manufacturing: Monitoring, Control and a Framework of Automated System." *Journal of Manufacturing Systems* 57: 31–45. doi:10.1016/j.jmsy.2020.08.008.

Xu, Fangda, Vimal Dhokia, Paul Colegrove, Anthony McAndrew, Stewart Williams, Andrew Henstridge, and Stephen T. Newman. 2018. "Realisation of a Multi-sensor Framework for Process Monitoring of the Wire Arc Additive Manufacturing in Producing Ti-6Al-4V Parts." *International Journal of Computer Integrated Manufacturing* 31 (8): 785–798. doi:10.1080/0951192X.2018.1466395.

Yuan, Lei, Zengxi Pan, Donghong Ding, Fengyang He, Stephen van Duin, Huijun Li, and Weihua Li. 2020. "Investigation of Humping Phenomenon for the Multi-directional Robotic Wire and Arc Additive Manufacturing." *Robotics and Computer-Integrated Manufacturing* 63: Article ID 101916. doi:10.1016/j.rcim.2019.101916.

Zhang, Zhifen, Guangrui Wen, and Shanben Chen. 2017. "Audible Sound-based Intelligent Evaluation for Aluminum Alloy in Robotic Pulsed GTAW: Mechanism, Feature Selection, and Defect Detection." *IEEE Transactions on Industrial Informatics* 14 (7): 2973–2983. doi:10.1109/TII.9424.

Zhao, Yu, Jun Wang, Qiujun Lu, and Ruise Jiang. 2010. "Pattern Recognition of Eggshell Crack Using PCA and LDA." *Innovative Food Science & Emerging Technologies* 11 (3): 520–525. doi:10.1016/j.ifset.2009.12.003.