Proceedings of the ASME 2021
International Design Engineering Technical Conferences &
Computers and Information in Engineering Conference
IDETC/CIE 2021
August 17- 20, 2021, Virtual, Online

DETC2021-69573

A STUDY ON THE ACOUSTIC SIGNAL BASED FRAMEWORKS FOR THE REAL-TIME IDENTIFICATION OF GEOMETRICALLY DEFECTIVE WIRE ARC BEAD

Nowrin Akter Surovi
Engineering Product Development
Singapore University of
Technology and Design
Singapore 487372
Email: surovi_akter@mymail.sutd.edu.sg

Audelia G. Dharmawan
Engineering Product Development
Singapore University of
Technology and Design
Singapore 487372
Email: audelia@sutd.edu.sg

Gim Song Soh
Engineering Product Development
Singapore University of
Technology and Design
Singapore 487372
Email: sohgimsong@sutd.edu.sg

ABSTRACT

In Wire Arc Additive Manufacturing (WAAM), weld beads are deposited bead-by-bead and layer-by-layer, leading to the final part. Thus, the lack of uniformity or geometrically defective bead will subsequently lead to voids in the printed part, which will have a great impact on the overall part quality and mechanical strength. To resolve this, several techniques have been proposed to identity such defects using vision or thermal-based sensing, so as to aid in the implementation of in-situ corrective measures to save time and cost. However, due to the environment that they are operating in, these sensors are not an effective way of picking up irregularities as compared to acoustic sensing. Therefore, in this paper, we seek to study into three acoustic feature-based machine learning frameworks - Principal Component Analysis (PCA) + K-Nearest Neighbors (KNN), Mel Frequency Cepstral Coefficients (MFCC) + Neural Network (NN) and Mel Frequency Cepstral Coefficients (MFCC) + Convolutional Neural Network (CNN) and evaluate their performance for the real-time identification of geometrically defective weld bead. Experiments are carried out on stainless steel (ER316LSi), bronze (ERCuNiAl) and mixed dataset containing both stainless steel and bronze. The results show that all three frameworks outperform the state-of-the-art acoustic signal based ANN approach in terms of accuracy. The best performing framework PCA+KNN outperforms ANN by more than 15%, 30% and 30% for stainless steel, bronze and mixed datasets, respectively.

1 INTRODUCTION

Wire Arc Additive Manufacturing (WAAM) is an arc-welding-based Direct Energy Deposition additive manufacturing technique. Recently, WAAM is becoming popular in the metal manufacturing industry because of its low equipment cost, low buy-to-fly ratio and high deposition rate [1, 2]. A WAAM product consists of multiple layers of overlapping weld beads. During the printing process, defects such as porosity, cracks, distortion, oxidation etc., may occur in the weld beads because of poor programming strategy, excessive heat accumulation, violation of the welding process, environmental influence and other machine malfunctions [3, 4]. Hence, it is important to identify the defective welding process early on so as to ensure the quality and consistency of the final printed WAAM product. In this paper, we focus on detecting defects in terms of the non-uniformity of the geometric shape of the weld beads, hereby referred to as *geometric defects* [5]. This is because irregular weld beads affect the strength [6] and quality [7] of the WAAM product as they contribute to voids or porosity within the part itself (See Figure 1). Therefore, real-time identification of geometrically defective beads during the printing process is essential so that appropriate corrective measures can be taken during the print process to save welding resources and material costs.

In the literature, Jia et al. [8], and Zhang et al. [9] used visual-based sensors to identify geometric defects in real-time through the use of support vector machine (SVM) and convolu-

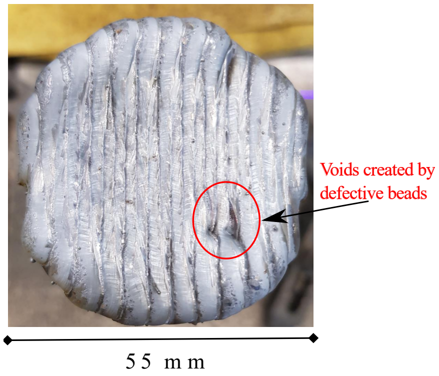


FIGURE 1: Geometrically defective beads lead to voids between two successive beads will affect the final printed part quality and strength.

tional neural network (CNN), respectively. Kryukov et al. [10] used a thermal camera to identify surface defects by studying the intensity profile during the friction weld process. However, the downside of the use of such sensors for our purpose is that they are light-sensitive, dust-sensitive, expensive and require high maintenance. On the other hand, acoustic sensors have several advantages over these sensors due to their low cost, easy to maintain, simple structure, highly targeted sensitive [11] and radiation free [1, 5]. Hence, this motivates us to explore acoustic-signal-based frameworks to identify geometrically defective beads.

There are several works that explored the relationships between the geometry of beads and the properties of acoustic signals [12]. Polajnar et al. [5] showed that irregularities in the bead geometry are reflected in the intensity of acoustic signals. Lv et al. [13] showed that the height of the arcs constituting the shape of the bead has a linear relationship with the pressure of the acoustic signal. Due to above-mentioned relationships between the properties of the bead geometry and the measured acoustic signals, this inspires us to extract and use acoustic features to capture the geometric defects during the WAAM process with appropriate feature-based models.

Acoustic features and acoustic feature-based models have been used in domains ranging from food science to voice recognition, and to medical diagnostic for solving various domain-specific classification problems. In the domain of food science and technology, Zhao et al. [14] extracted features from the time-domain acoustic signal and applied Principal Component Analysis (PCA) to determine eggshell crack. Another approach is through the use of Mel-Frequency Cepstral Coefficients (MFCCs) to extract features [15], and combined with Neural Network (NN) [16] and Convolutional Neural Network (CNN) [17] for classification. Such approaches are commonly applied in voice recognition problem. In the medical domain, Zabidi et al. [18] used MFCC features and CNN for detecting infant asphyxia disease based on their crying sound. Encouraged by the success of acoustic features and acoustic feature-based models in the above domains, our goal here seek to extend this into the additive manufacturing domain through the use of acoustic signal based framework for identifying defective beads. In the domain of additive manufacturing, there is an acoustic signal based methodology that uses Artificial Neural Network (ANN) [19] to identify defective beads. ANN uses time domain acoustic signals as features to train a neural network.

In this paper, we propose and evaluate three different acoustic signal based frameworks for geometrically defective bead identification: PCA + KNN, MFCC + NN and MFCC + CNN. PCA + KNN (Section 3.2) uses Principal Component Analysis (PCA) for feature extraction and K-Nearest Neighbors (KNN) for identifying the detective beads. MFCC + NN (Section 3.3) uses Mel Frequency Cepstral Coefficients (MFCCs) as features and trains a Neural Network (NN) using those features for identification of the defective beads. MFCC + CNN (Section 3.4) converts the MFCCs features into images and trains Convolutional Neural Network (CNN) using those images for identification of the defective beads. To understand the suitability of these frameworks for our purpose across different materials, we empirically evaluate these frameworks on two types of printed materials, namely, bronze and stainless steel (Section 2 and Section 4).

We comparatively evaluate our frameworks with the existing ANN framework [19]. Our frameworks, particularly the MFCC+NN and MFCC+CNN, differ from ANN in that our frameworks use spectral features (MFCCs), whereas ANN uses time-domain signals. Our frameworks are architecturally different from ANN and furthermore optimised for performance (see Section 3.3 and Section 3.4 for details). The empirical evaluation suggests that the best-performing framework PCA+KNN outperforms the existing ANN by more than 15%, 30% and 30% for stainless steel, bronze and mixed datasets respectively (Section 4.2).

2 EXPERIMENTAL SETUP AND DATA COLLECTION

In this section, we explain our experimental setup, data collection and dataset labelling.

Experimental setup The experiments were conducted on our robotic WAAM system at Singapore University of technology and Design (SUTD) as shown in Figure 2. 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), a 2D laser scanner (Micro-Epsilon scanCONTROL 2910-100) and a microphone (Hoco M19 DRUMBEAT) that is connected around 60 cm above the substrate in order to

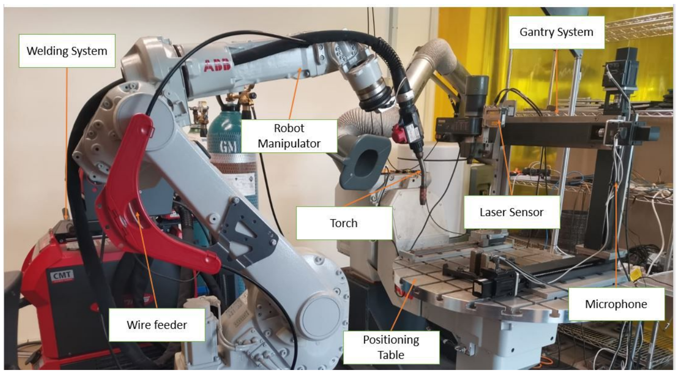


FIGURE 2: Experimental setup of SUTD robotic WAAM for Data Collection

minimize environmental noise. The gantry system is controlled to move the laser scanner in 3D space in order to obtain 3D point clouds of the printed weld beads.

Data collection We printed 26 weld beads using bronze (ERCuNiAl) wires and 52 weld beads using stainless steel (ER316LSi) wires on different combination of torch speed and wire feed rate in order to obtain different acoustic signals and weld bead geometry. The torch speed and wire feed rate were in the range of [3,10] mm/s and [3,8] m/min respectively for bronze as well as [3,15] mm/s and [3,6] m/min respectively for stainless steel.

We collected 26 bronze signals and 52 stainless steel signals. The signals were collected at 8KHz sampling rate during the welding process and stored in our computer. We segmented each of the collected signals into ten signal segments. Thus, the *bronze dataset* contains 260 bronze signal segments and *stainless steel dataset* contains 520 stainless steel signal segments.

Dataset labelling Our objective is to identify geometrically defective beads. In order to train an acoustic signal based model to identify a defective bead, we need to assign labels to the signal segments based on which the model would learn to make decisions. Hence we label each of the signal segments in the bronze and stainless steel datasets.

In order to label signal segments, we first determine the uniformity of the beads. To do so, we measured the height, width and area of each bead using a moving 2D laser scanner. Then we calculated the Root Mean Square Error (RMSE) value of the height, width and area of each bead. Finally, we calculated a combined RMSE by

$$RMSE = \left(\frac{width_{RMSE} + height_{RMSE} + \sqrt{area_{RMSE}}}{3}\right) \qquad (1)$$

A bead with a smaller combined RMSE is more uniform; we consider it a good (non-defective) bead. Similarly, A bead with a larger combined RMSE is less uniform; we consider it a bad (defective) bead. We find that the threshold for a Stainless steel defective bead is 0.377mm and a Bronze defective bead is 0.5mm. Examples of geometrically defective and non-defective beads and their corresponding acoustic waveforms are shown in Figure 3 and Figure 4.

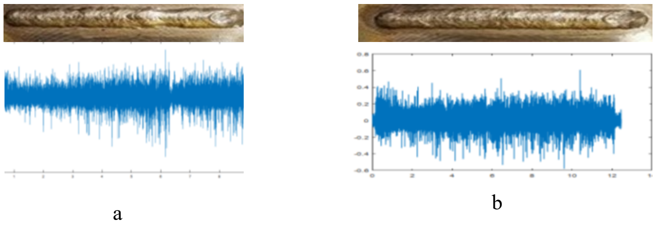


FIGURE 3: Geometrically defective and non-defective beads with acoustic time-domain waveforms (a) Bronze with defects ($RMSE = 0.62$mm) (b) Bronze without defects ($RMSE = 0.44$mm)

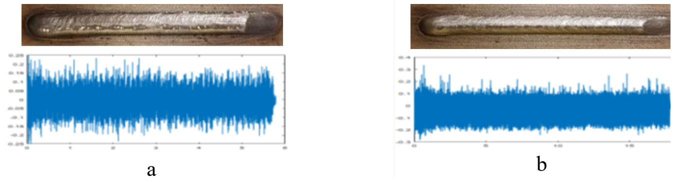


FIGURE 4: Geometrically defective and non-defective beads with acoustic time-domain waveforms (a) Stainless steel with defects ($RMSE = 0.409$mm) (b) Stainless steel without defects ($RMSE = 0.27$mm)

All the signal segments of a good bead are labelled good, and all the signal segments of a bad bead are labelled bad. In this manner, we have labelled around 40% bronze signal segments as bad and 60 % bronze signal segments as good, while 20% stainless steel signal segments as bad and 80 % stainless steel signal segments as good beads. In the mixed dataset, around 25% signal segments are labelled bad, and 75 % signal segments are labelled good.

3 DESCRIPTION OF THE ACOUSTIC SIGNAL BASED FRAMEWORKS

In this section, we explain our proposed frameworks for feature extraction and identification of geometrically defective beads are as shown in Figure 5. It begins with Downsampling, followed by Feature Extraction and then by Identification.

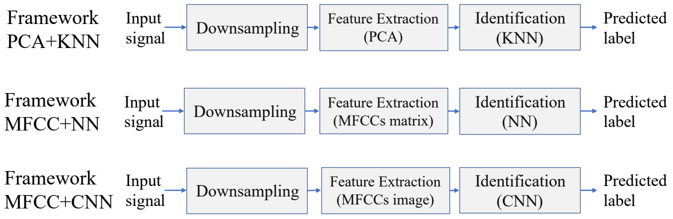


FIGURE 5: Our proposed frameworks for identification of defective beads

3.1 Downsampling

As we use different torch speed and wire feed rate for printing the similar length beads, the number of acoustic signal data points of each bead is different. Therefore, we use downsampling approach to make the same number of data points for all acoustic signals so that the same number of data points for each set can be subsequently used for feature extraction. Let us consider, z^i as the acoustic signal of the i^{th} bead, T^i as the number of data points in z^i, and T_{min} as the number of data points in the smallest signal. If there is a positive integer k that satisfies $k * T_{min} \leq T^i < (k+1) * T_{min}$, then the downsampled signal as denoted by x^i is given by the following sequence

$$x^i = (z^i(1), z^i(1+k), z^i(1+2k), z^i(1+3k), \dots) \qquad (2)$$

For example, if $T_{min} \leq T^i < 2T_{min}$, we downsample by selecting the first T_{min} data points from z^i. If $2T_{min} \leq T^i < 3T_{min}$, we downsample by selecting 1^{st}, 3^{rd}, 5^{th} data points, and so on from z^i.

3.2 PCA+KNN

In this section, we explain the feature extraction and identification procedure for PCA+KNN framework.

Feature Extraction (PCA) PCA is a dimension reduction technique which reduces the dimensions of the data while keeping most of the information of the original data. PCA uses singular value decomposition and projects the high dimensional data into a lower dimension space [20]. Our PCA feature extraction procedure is given below:

1. Compute Discrete Fourier Transform (DFT) [21] on the downsampled signal using the following equation:

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j2\pi nk/N}, \quad 0 \leq k \leq (N-1) \qquad (3)$$

where N is the number of points used to compute the DFT.

2. Use wavelet threshold denoising [22] to denoise the DFT signal to get y(k), and assemble them into a matrix. $[Y] = \{y(0), y(2), \dots, y(N-1)\}$.
3. Calculate the covariance matrix $[Q] = [Y][Y]^T$.
4. Calculate the eigenvectors and eigenvalues of the covariance matrix using the following equation: $\lambda_i e_i = [Q]e_i$ where λ_i is the eigenvalue associated with the eigenvector e_i.
5. Sort the eigenvectors in the descending order of eigenvalues. The eigenvectors with the highest, the second-highest eigenvalues form the 1^{st}, 2^{nd} principal components and so on.

Identification (KNN) KNN is a supervised learning technique. The algorithm works by selecting the K^{th} nearest points of a new example by calculating the distance among all existing examples with the new example. The K^{th} neighbors are determined by the shortest distance from the new example. The class of new example is assigned to the majority class of K^{th} nearest points [23]. For this application, we find that $K = 3$ (3 nearest neighbors) is ideal as it gives minimum validation loss and with the first six principal components is sufficient for training the KNN to yield maximum validation accuracy.

3.3 MFCC+NN

In this section, we explain the feature extraction and identification procedure for MFCC+NN framework.

Feature Extraction (MFCC) MFCCs is a popular acoustic feature extraction technique and is used for speech recognition [24] and emotion recognition [25]. Mel frequency here refers to human audible range frequency. MFCCs are short term power spectrum based features [26] which capture the distinguishing characteristics of sound. Our MFCCs feature extraction procedure [21] from downsampled acoustic signal is given below:

1. Convert each segmented signal into frames.
2. On each frame, apply a window function (e.g. Hamming window) to get a windowed signal.
3. Compute Discrete Fourier Transform (DFT) on the windowed signal using Eq. (3).
4. Compute Mel-filter bank, which is a set of band-pass filters. The filter-bank is a nonlinear-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 MFCCs computation. The conversion of physical frequency to Mel frequency is given below:

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

where f is the physical frequency in Hz and f_{Mel} is the human perceived frequency.

5. Calculate filter-bank energies by multiplying the square of the magnitude spectrum of DFT signal with each filter-bank:

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

where M is the total number of triangular Mel filters and $H_m(k)$ is the weight given to the k^{th} energy spectrum bin contributing to the m^{th} output band.
6. Calculate the logarithm of the energies.
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 of getting coefficients is given below:

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

where $n = 0, 1, 2, ..., C-1$, c(n) are the cepstral coefficients and C is the number of MFCCs. The coefficients of each frame form a row vector.
8. Construct a matrix (*MFCCs feature matrix*) for each segmented signal using the row vectors of the frames.

Identification (NN) The architecture of our proposed NN consists of one input layer, three hidden layers and one output layer, as shown in Figure 6. Inputs to the NN are the MFCCs feature matrices of all segmented signals. The hidden layers contain 512, 256 and 64 neurons, respectively. Rectified Linear Unit (ReLU) activation function [27] is used in all hidden layers. L_2 regularization (0.001) [28] is used in every hidden layer to avoid overfitting problem.

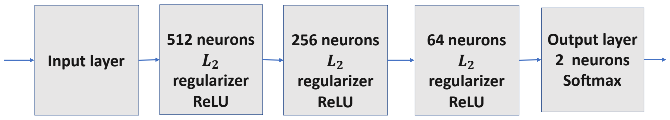
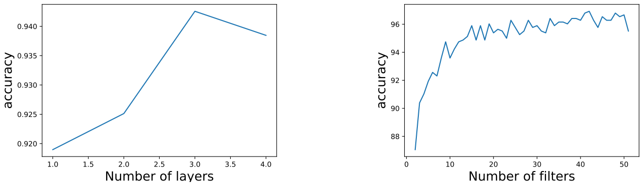


FIGURE 6: Our NN Architecture for defective bead Identification

The reason behind using such an architecture is that, it is optimised for performance. To be specific, we vary the number of hidden layers and select such a number of hidden layers that provides the highest testing accuracy on mixed dataset. As shown in Figure 7 (a), a 3-layer NN provides the maximum classification accuracy.



(a) NN layer choice

(b) CNN filter choice

FIGURE 7: NN and CNN architecture choice

3.4 MFCC+CNN

In this section, we explain the feature extraction and identification procedure for MFCC+CNN framework.

Feature Extraction (MFCC) First, we follow the same procedure explained in Section 3.3 for extracting feature matrix of a segmented signal. Finally, we convert each MFCCs feature matrix of a segmented signal into an image. We refer this image as *MFCCs feature image*. An example of MFCCs feature image is shown in Figure 8.

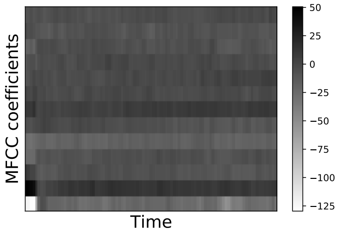


FIGURE 8: MFCCs feature image of a segmented signal

Identification (CNN) The architecture of our proposed CNN for defective bead identification is made up of one convolutional layer, one pooling layers, one fully connected layer, and one output layer as shown in Figure 9. Inputs to the CNN are the MFCCs feature images of all segmented signals. The convolutional layer

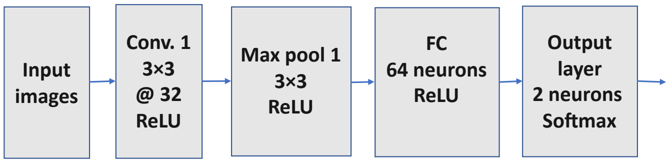


FIGURE 9: Our CNN Architecture for defective bead Identification

consists of 32 filters, and the size of each filter is 3 × 3. Also, the max-pooling layer filter size 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 choice of 32 filters in the convolutional layer is in order to achieve a better classification accuracy. To be specific, we vary the number of filters in the convolutional layer from 1 up to 50 and using 10-fold cross validation approach we select such a number of filters that provides the highest testing accuracy on mixed dataset. As shown in Figure 7 (b), we observe that testing accuracy is quite similar after around 32 filters. Therefore, we select 32 filters in the convolutional layer.

4 EXPERIMENTAL RESULTS AND DISCUSSION

In this section, we report and evaluate the training mechanism (Section 4.1), performance (Section 4.2) and robustness (Section 4.3) of the proposed frameworks.

4.1 Training Mechanism

In this section, we report the training procedures of the frameworks and discuss choices of parameters (if any) for the purpose of demonstrating soundness and reproducibility of the models used later for other empirical studies.

TABLE 1: NUMBER OF TRAINING, VALIDATION AND TESTING SAMPLES OF DIFFERENT DATASETS

Dataset	Training	Validation	Testing
Bronze	156	39	65
Stainless Steel	312	78	130
Mixed	468	117	195

PCA+KNN We extract the 1st 25 principal components from bronze and stainless steel datasets. In order to validate the minimum number of principal components required to train the KNN, we split the mixed dataset randomly as 60% for training, 25% for testing and 15% for validation as shown in Table 1. Then we calculate the validation accuracy for different components. We follow the similar process 30 times and get mean validation accuracy for different components and, plot in Figure 10. We observe that the mean validation accuracy increases at the beginning and reach around 97 % for the first six components and then the accuracy become stable. Therefore, training the first six principal components is enough to get maximum validation accuracy.

Thus utilizing the first six principal components and after training our KNN 30 times randomly, we get training and validation accuracy at around 98% and 97% respectively for the mixed dataset. When we split the mixed dataset into their respective materials, we get training and validation accuracy at around 96% and 91% for the bronze dataset, and 99% and 99% for stainless steel dataset respectively.

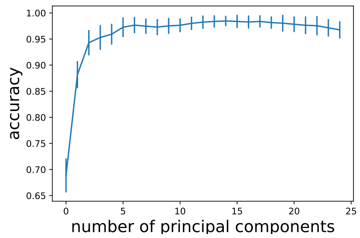


FIGURE 10: Accuracy vs number of principal components

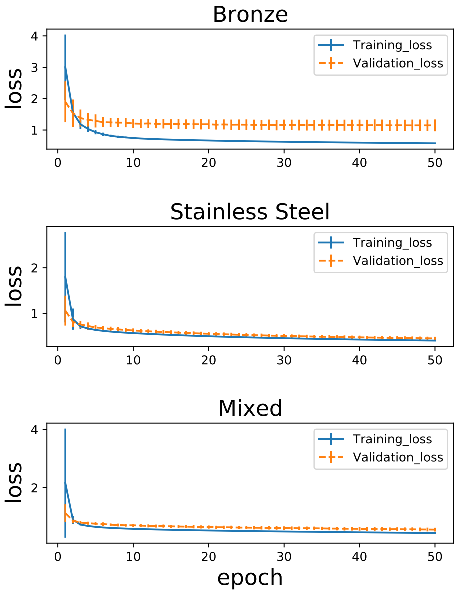


FIGURE 11: Learning curves of MFCC + NN on different datasets

MFCC + NN After extracting the MFCCs feature matrices, we use the features for training the NN. We split the mixed dataset as 60% for training, 25% for testing and 15% for validation. Adam optimizer [29] with sparse categorical cross-entropy loss is used for training the NN and the value of 0.0001 is used as the learning rate. We run 50 iterations for each training and validation. We also train the network 30 times to check for its stability. Thus using this framework, we get training and validation accuracy at around 99% and 95% respectively for the mixed dataset. Similarly, we get training and validation accuracy at around 99 % and 89% for the bronze dataset as well as 99 % and 96 % for the stainless steel dataset respectively.

The learning curves of MFCC+NN on the bronze dataset, stainless steel dataset and mixed dataset are shown in Figure 11. In the curves, we observe that both training and validation loss decrease as we increase the number of epochs. After 50 epochs, both training and validation loss become stable with a minimum gap. Therefore, we conclude that the MFCC+NN framework has been trained without over-fitting.

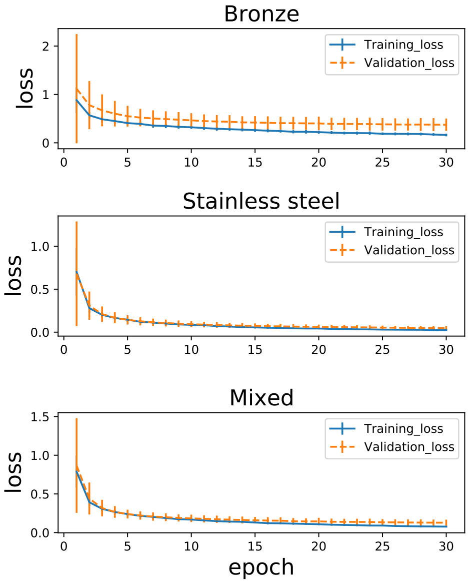


FIGURE 12: Learning curves of MFCC+CNN on different datasets

MFCC + CNN Similarly, we use the MFCCs feature images for training the CNN. We split the mixed dataset as 60% for training, 25% for testing and 15% for validation. We use Adam optimizer [29] with sparse categorical cross-entropy loss for training the CNN and 0.0001 as the value of the learning rate. We run 30 iterations for each training and validation. We train the network 30 times to check its stability. After training, we get training and validation accuracy at around 97% and 94% respectively for the mixed dataset. Again, we get training and validation accuracy at around 96 % and 88% for the bronze dataset as well as 99 % and 97 % for stainless steel dataset separately.

The learning curves of MFCC+CNN on the bronze dataset, stainless steel dataset and mixed dataset are shown in Figure 12. In the curves, we observe that both training and validation loss decrease as we increase the number of epochs. After 30 epochs, both training and validation loss become stable with a minimum gap. Therefore, we conclude that the MFCC+CNN framework has been trained without overfitting.

Discussion We observe from the training mechanism for all frameworks that validation accuracy for the stainless steel dataset is always higher than that of bronze dataset. This result can be explained by observing that stainless steel training size is larger than that of bronze (Table 1) and by the fact that testing accuracy increases with an increase in training sample size [30].

Although the training size of the mixed dataset is larger than that of other datasets, the validation accuracy of the mixed dataset is not the largest. We believe the reason is that the bronze and stainless steel datasets have different distributions. To verify that the two datasets have different distributions, we have trained our frameworks on the stainless steel before testing on the bronze, and conversely, trained the frameworks on bronze before testing on stainless steel. We have found the testing accuracy to be less than 25 % if we train our framework on stainless steel and tested on bronze, and 60 % vice versa.

4.2 Performance Evaluation

To evaluate the performance of the proposed frameworks, we calculate the mean and standard deviation of testing accuracy, construct the confusion matrices, and measure the F1 score for all frameworks. We report the testing accuracy in Table 2, confusion matrices in Table 3 and F1 score in Table 4 respectively.

Table 2 shows a comparative study of our proposed frameworks as well as its comparison with the state-of-art ANN model [19]. This is based on single material data sets of Bronze and Stainless Steel, and mixed dataset with both materials combined. From the table, we observe that our proposed frameworks outperform the ANN framework in terms of accuracy. Furthermore, we observe that PCA+KNN has the highest accuracy among all the frameworks.

In order to explain the comparatively lower accuracy of MFCC+NN and MFCC+CNN, we vary the training size and measure the testing accuracy of the frameworks as shown in Figure 13. We observe that PCA+KNN yield a higher accuracy than other frameworks for the same training size. Furthermore, MFCC+NN and MFCC+CNN require a higher training size to achieve the same accuracy as that of PCA+KNN. For instance, the black straight lines in Figure 13, show that a training size of 300 is sufficient for PCA+KNN to achieve 94.5% testing accuracy, while MFCC+NN and MFCC+ CNN require training sizes of around 350 and 450 to achieve the same. This implies that our dataset size (as shown in Table 1) is not large enough for the deep learning based frameworks to perform well and we will investigate more into this aspect as part of future work.

We observe that our datasets are imbalanced. Therefore, we measure the confusion matrix, recall, precision, and F1 scores to evaluate our frameworks. The reasons are two-fold. First, it

TABLE 2: COMPARATIVE STUDY OF OUR FRAMEWORKS VS EXISTING APPROACH ANN [19]

Dataset	**Method**	**Mean Accuracy (test-set)**	**Std Accuracy (test-set)**
Bronze	PCA+KNN	**90.09 %**	**0.0363**
	MFCC+NN	**89.69 %**	**0.0328**
	MFCC+CNN	**86.47 %**	**0.0394**
	ANN	60.78 %	0.0620
Stainless Steel	PCA+KNN	**98.99 %**	**0.0923**
	MFCC+NN	**97.84 %**	**0.0151**
	MFCC+CNN	**97.01 %**	**0.0236**
	ANN	84.46 %	0.03327
Mixed	PCA+KNN	**96.37 %**	**0.0571**
	MFCC+NN	**95.85 %**	**0.0132**
	MFCC+CNN	**94.99 %**	**0.0153**
	ANN	64.94%	0.1286

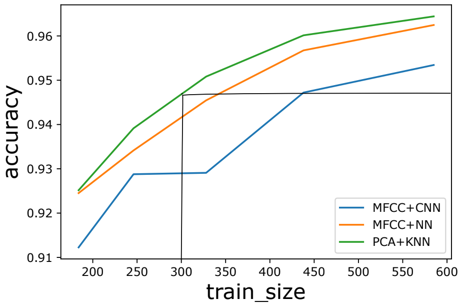


FIGURE 13: Accuracy vs training size (mixed) of all frameworks

serves as a quality assessment for a framework with an imbalanced dataset. Second, it allows us to check the performance of our frameworks. From the confusion matrices in Table 3, we observe that despite our datasets being imbalanced, our frameworks can effectively pick out good and bad segments. In comparison, we observe that the existing ANN framework shows a significant number of false negatives, meaning wrongly identifying a bad bead as a good bead. Furthermore, as shown in Table 4, we observe that the precision, recall and F1 score values of our frameworks are higher than that of the existing framework (ANN). Therefore, we can conclude that our frameworks perform well even with an imbalanced dataset.

Next, we also measure the computation speed of the universal frameworks PCA+KNN, MFCC+NN and MFCC+CNN, and they are found to be less than 50 ms for feature extraction and defective bead identification. This seems to be acceptable to be used in a real-time situation.

TABLE 3: CONFUSION MATRICES OF ALL FRAMEWORKS ON DIFFERENT DATASETS

Frameworks			**True**	
			Good	Bad
PCA+KNN	**Predicted**	Good	35	3
		Bad	2	25
MFCC+NN	**Predicted**	Good	36	4
		Bad	1	24
MFCC+CNN	**Predicted**	Good	36	3
		Bad	1	25
ANN	**Predicted**	Good	37	27
		Bad	0	1

BRONZE

Frameworks			**True**	
			Good	Bad
PCA+KNN	**Predicted**	Good	108	1
		Bad	0	21
MFCC+NN	**Predicted**	Good	107	2
		Bad	1	20
MFCC+CNN	**Predicted**	Good	108	1
		Bad	0	21
ANN	**Predicted**	Good	108	22
		Bad	0	0

STAINLESS STEEL

Frameworks			**True**	
			Good	Bad
PCA+KNN	**Predicted**	Good	153	3
		Bad	0	39
MFCC+NN	**Predicted**	Good	151	3
		Bad	2	39
MFCC+CNN	**Predicted**	Good	148	5
		Bad	5	37
ANN	**Predicted**	Good	152	41
		Bad	1	1

MIXED

TABLE 4: COMPARATIVE STUDY OF ALL FRAMEWORKS

Data-set	Method	precesion	Recall	F1-Score
Bronze	PCA+KNN	91 %	91 %	91 %
	MFCC+NN	84%	83 %	83 %
	MFCC+CNN	88 %	88 %	88 %
	ANN	34 %	58 %	43 %
Stainless Steel	PCA+KNN	99 %	99 %	99 %
	MFCC+NN	99 %	99 %	99 %
	MFCC+CNN	99 %	98 %	99 %
	ANN	78 %	88 %	83 %
Mixed	PCA+KNN	98 %	98 %	98 %
	MFCC+NN	97 %	97 %	97 %
	MFCC+CNN	95 %	95 %	95 %
	ANN	85 %	81 %	73 %

4.3 Robustness

Note that labels of our beads can be slightly inaccurate because the thresholds are based on human perception and can be subjective. However, it is not our focus to solve the mislabelling problem in this paper. Nevertheless, a way to address this problem is to check for the robustness of our frameworks with respect to mislabelling. Since our proposed frameworks give higher accuracy than ANN, we went ahead to evaluate the robustness of our proposed frameworks.

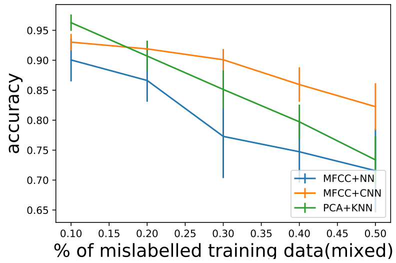


FIGURE 14: Robustness of PCA+KNN, MFCC+NN and MFCC+CNN frameworks with respect to training set mislabelling

To evaluate the robustness, we split the mixed dataset uniformly at random into 75% training set and 25% testing set. We corrupt our training label from 10% up to 50% and calculate the mean testing accuracy based on the original labels of the test dataset. As shown in the results in Figure 14, we observe that we get above 75 %, 70% and 80% mean testing accuracy against 50% mislabelled train dataset for PCA+KNN, MFCC+NN and MFCC+CNN frameworks, respectively. Therefore, we can conclude MFCC+CNN is more robust than others.

5 CONCLUSIONS

This paper presents a study on the performance of real-time sensing of geometrically defective beads using acoustic-signal-based frameworks in the robotic WAAM process. We have proposed and evaluated three universal identification frameworks, namely PCA+KNN, MFCC+NN and MFCC+CNN, for sensing geometric defects. Among these frameworks, PCA+KNN performs best in terms of accuracy, while MFCC+CNN performs best in terms of robustness. These findings serve as our first step in developing an intelligent system for the identification of defective beads early on the printing process so that appropriate intervention could be implemented to save welding resources and material costs.

In future, we want to tackle the issues of class imbalance and labelling errors due to human perception in our datasets. Moreover, we also plan to characterize the different types of geometric defects such as Blowout, Undercut, Humping etc. and explore their relationship with acoustic signals so as to further enhance the defect identification of our WAAM system.

REFERENCES

[1] Xu, F., Dhokia, V., Colegrove, P., McAndrew, A., Williams, S., Henstridge, A., and Newman, S. T., 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), pp. 785–798.

[2] Busachi, A., Erkoyuncu, J., Colegrove, P., Martina, F., and Ding, J., 2015. "Designing a waam based manufacturing system for defence applications". *Procedia Cirp*, ***37***, pp. 48–53.

[3] Kawahito, Y., Matsumoto, N., Abe, Y., and Katayama, S., 2011. "Relationship of laser absorption to keyhole behavior in high power fiber laser welding of stainless steel and aluminum alloy". *Journal of Materials Processing Technology*, ***211***(10), pp. 1563–1568.

[4] Wu, B., Pan, Z., Ding, D., Cuiuri, D., Li, H., Xu, J., and Norrish, J., 2018. "A review of the wire arc additive manufacturing of metals: properties, defects and quality improvement". *Journal of Manufacturing Processes*, ***35***, pp. 127–139.

[5] Polajnar, I., Bergant, Z., and Grum, J., 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, pp. 613–20.

[6] Rajashekar, R., and Rajaprakash, B., 2016. "Development of a model for friction stir weld quality assessment using machine vision and acoustic emission techniques". *Journal of Materials Processing Technology,* ***229***, pp. 265–274.

[7] Cayo, E., and Alfaro, S., 2012. "Real-time assessment of the stability of metal transfer in gmaw-s process based on arc emissions". In The 6th International Conference—Innovative technologies for joining advanced materials, Vol. 1, pp. 1–4.

[8] Jia, H., Murphey, Y. L., Shi, J., and Chang, T.-S., 2004. "An intelligent real-time vision system for surface defect detection". In Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004., Vol. 3, IEEE, pp. 239–242.

[9] Zhang, Y., You, D., Gao, X., Zhang, N., and Gao, P. P., 2019. "Welding defects detection based on deep learning with multiple optical sensors during disk laser welding of thick plates". *Journal of Manufacturing Systems,* ***51***, pp. 87–94.

[10] Kryukov, I., Hartmann, M., Böhm, S., Mund, M., Dilger, K., and Fischer, F., 2014. "Defect detection in friction stir welding by online infrared thermography". *Journal of Welding and Joining,* ***32***(5), pp. 480–487.

[11] Kaushik, B., Nance, D., and Ahuja, K., 2005. "A review of the role of acoustic sensors in the modern battlefield". In 11th AIAA/CEAS Aeroacoustics Conference, p. 2997.

[12] Heralic, A., 2012. *Monitoring and control of robotized laser metal-wire deposition*. Chalmers University of Technology.

[13] Lv, N., Zhong, J., Chen, H., Lin, T., and Chen, S., 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), pp. 235–249.

[14] Zhao, Y., Wang, J., Lu, Q., and Jiang, R., 2010. "Pattern recognition of eggshell crack using pca and lda". *Innovative Food Science & Emerging Technologies,* ***11***(3), pp. 520–525.

[15] Muda, L., KM, B., and Elamvazuthi, I., 2010. "Voice recognition algorithms using mel frequency cepstral coefficient (mfcc) and dynamic time warping (dtw) techniques". *Journal of Computing,* ***2***(3), pp. 138–143.

[16] Tanweer, S., Mobin, A., and Alam, A., 2014. "Analysis of combined use of nn and mfcc for speech recognition". *International Journal of Computer, Electrical, Automation, Control and Information Engineering,* ***8***(9).

[17] Palaz, D., Collobert, R., et al., 2015. Analysis of cnn-based speech recognition system using raw speech as input. Tech. rep., Idiap.

[18] Zabidi, A., Yassin, I., Hassan, H., Ismail, N., Hamzah, M., Rizman, Z., and Abidin, H. Z., 2017. "Detection of asphyxia in infants using deep learning convolutional neural network (cnn) trained on mel frequency cepstrum coefficient (mfcc) features extracted from cry sounds". *Journal of Fundamental and Applied Sciences,* ***9***(3S), pp. 768–778.

[19] Pernambuco, B. S. G., Steffens, C. R., Pereira, J. R., Werhli, A. V., Azzolin, R. Z., and Estrada, E. d. S. D., 2019. "Online sound based arc-welding defect detection using artificial neural networks". In 2019 Latin American Robotics Symposium (LARS), 2019 Brazilian Symposium on Robotics (SBR) and 2019 Workshop on Robotics in Education (WRE), IEEE, pp. 263–268.

[20] Tipping, M. E., and Bishop, C. M., 1999. "Probabilistic principal component analysis". *Journal of the Royal Statistical Society: Series B (Statistical Methodology),* ***61***(3), pp. 611–622.

[21] Rao, K. S., and Manjunath, K., 2017. *Speech Recognition Using Articulatory and Excitation Source Features*. Springer.

[22] Zhang, Z., Wen, G., and Chen, S., 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), pp. 2973–2983.

[23] Rivero, D., Fernandez-Blanco, E., Dorado, J., and Pazos, A., 2011. "A new signal classification technique by means of genetic algorithms and knn". In 2011 IEEE Congress of Evolutionary Computation (CEC), IEEE, pp. 581–586.

[24] Ittichaichareon, C., Suksri, S., and Yingthawornsuk, T., 2012. "Speech recognition using mfcc". In International Conference on Computer Graphics, Simulation and Modeling, pp. 135–138.

[25] Sato, N., and Obuchi, Y., 2007. "Emotion recognition using mel-frequency cepstral coefficients". *Information and Media Technologies,* ***2***(3), pp. 835–848.

[26] Logan, B., et al., 2000. "Mel frequency cepstral coefficients for music modeling.". In Ismir, Vol. 270, Citeseer, pp. 1–11.

[27] Nair, V., and Hinton, G. E., 2010. "Rectified linear units improve restricted boltzmann machines". In ICML.

[28] Cortes, C., Mohri, M., and Rostamizadeh, A., 2012. "L2 regularization for learning kernels". *arXiv preprint arXiv:1205.2653*.

[29] Kingma, D. P., and Ba, J., 2014. "Adam: A method for stochastic optimization". *arXiv preprint arXiv:1412.6980*.

[30] Foody, G., and Arora, M., 1997. "An evaluation of some factors affecting the accuracy of classification by an artificial neural network". *International Journal of Remote Sensing,* ***18***(4), pp. 799–810.