Interpretable & Explainable Machine Learning for Ultrasonic Defect Sizing

Richard J. Pyle, Robert R. Hughes, Paul D. Wilcox, *Member, IEEE*

***Abstract*—Despite its popularity in literature, there are few examples of machine learning (ML) being used for industrial nondestructive evaluation (NDE) applications. A significant barrier is the 'black box' nature of most ML algorithms. This paper aims to improve the interpretability and explainability of ML for ultrasonic NDE by presenting a novel dimensionality reduction method: Gaussian feature approximation (GFA). GFA involves fitting a 2D elliptical Gaussian function an ultrasonic image and storing the seven parameters that describe each Gaussian. These seven parameters can then be used as inputs to data analysis methods such as the defect sizing neural network presented in this paper. GFA is applied to ultrasonic defect sizing for inline pipe inspection as an example application. This approach is compared to sizing with the same neural network, and two other dimensionality reduction methods (the parameters of 6 dB drop boxes and principal component analysis), as well as a convolutional neural network applied to raw ultrasonic images. Of the dimensionality reduction methods tested, GFA features produce the closest sizing accuracy to sizing from the raw images, with only a 23% increase in RMSE, despite a 96.5% reduction in the dimensionality of the input data. Implementing ML with GFA is implicitly more interpretable than doing so with principal component analysis or raw images as inputs, and gives significantly more sizing accuracy than 6 dB drop boxes. Shapley additive explanations (SHAP) are used to calculate how each feature contributes to the prediction of an individual defect's length. Analysis of SHAP values demonstrates that the GFA-based neural network proposed displays many of the same relationships between defect indications and their predicted size as occur in traditional NDE sizing methods.**

***Index Terms*—Interpretability, machine learning ultrasound, defect characterization, neural network, plane wave imaging, simulation**

I. Introduction

Inferring the structural integrity of components without damaging them is an essential task for many industries, especially those with high-value or safety critical components. Non-destructive evaluation (NDE) techniques solve this challenge through analysis of a component's response to stimuli such as X-ray or ultrasound. Most NDE inspections must be carried out many times and usually produce very high-dimensional data, making manual data interpretation expensive. This motivates the use of automated data analysis, and as this is essentially a pattern recognition challenge, machine learning (ML) is well-suited. This has been shown repeatedly by the demonstration of human-level data interpretation performance both in NDE [1]–[9] as well as related fields such as computer vision [10] and medical imaging [11], [12].

This paper considers defect sizing for ultrasonic inline pipe inspection as an example industrial application. Earlier work for this application has shown that a convolutional neural network (CNN, [13]) can provide accurate defect sizing for experimental data, even when a simulated training set is used [9]. This drastically lowers the amount of experimental data required to implement ML based defect sizing. It has also been shown that domain adaptation can be used to further increase sizing accuracy if a small amount of experimental training data is available to supplement simulated training data [14] and effective uncertainty quantification is possible using a deep ensemble [15]. However, qualification of inspections using ML for safety critical applications such as pipeline inspection is still a challenge due to the 'black-box' nature of ML making it hard to build trust in, and certify, its predictions. This paper aims to tackle this issue by improving both the interpretability and local (i.e., for a specific test sample) explainability of ML based models for ultrasonic defect sizing. Note that in this paper the term *model* is used exclusively to describe any algorithm learnt from data, for example, a neural network, while *simulation* is used to describe a physics-based approach to approximating real data.

The precise definitions for *interpretability* and *explainability* are disagreed upon both between and within research fields. This work follows the definitions laid out in [16]. *Interpretability* is a domain specific notion, but in general it is the ability for a human to understand the link between cause and effect without anything other than the model itself. An explanation is an approximation of a model that aims to describe the cause of a local prediction. The term *explainability* is used here to follow convention but, as pointed out in [16], "summaries of predictions," "summary statistics," or "trends" are more truthful descriptors as the fact that "explanations" are an approximation to the complex internal calculations within a model is often overlooked.

Explanations for ML based on images are commonly provided by saliency maps which describe the locations in the input data that most significantly impact the prediction. There are many methods for creating saliency maps, such as gradient-weighted class activation mapping (grad-CAM, [17]), local interpretable model-agnostic explanations (LIME, [18]), deep learning important features (DeepLIFT, [19], [20]) and layer-wise relevance propagation (LRP, [21]). Shapley additive explanations (SHAP, [22]) provide a unified view of these methods, giving model-agnostic feature importance values for

any type of input data and any type of model. However, as pointed out in [16] a saliency map does not show what about that location in the image is important (e.g., texture/amplitude/color). In the authors' opinions this means saliency maps are of little use for most ML based defect classification where images or time domain signals are used as input, as highlighting the defect's indication in the data is of little use when it is usually already clear where the indication is. In other words, the challenge is interpreting how properties of an indication inform the prediction, rather than explaining which parts of an image led to the prediction. The root cause of this problem is a lack of interpretability in the model, due to the complex nature of the input data.

While interpretable ML is a relatively new field it has attracted a lot of research attention from the computer science community in recent years, due to its potential to address the 'black box' nature of ML [23]. However, within NDE there have been only a small number of publications with a focus on either explainable or interpretable ML. Saliency-map based explanations have been produced using LIME, for ultrasonic defect detection [24]. Text-based explanations have been used with a human-designed decision tree for crack characterization [25], an effective approach when the decision-making process of the model is simple enough to be explained in a small number of sentences. Improving the interpretability of ML methods for ultrasonic NDE data has been achieved by replacing the trainable convolutional filters of a CNN with filters matched to the shape of Lamb waves [26] in application to localizing damage in aluminum plate using guided waves. Another published approach is to use well-known dimensionality reduction methods such as principal component analysis (PCA) to reduce the complexity of input data. This has been used with a support vector machine (SVM) to detect damage in carbon fiber reinforced polymer plate using ultrasonic guided wave data [27].

As discussed in [28] it is important to consider what constitutes useful interpretability for the relevant domain when applying ML, as it can vary a lot between applications. In NDE, useful interpretability usually stems from the ability to relate a model's inner workings to the reasoning of a skilled human operator or a physics-based approach. Ensuring input data is of a reasonably low dimensionality is also essential for achieving this goal, as humans are not able to process high-dimensional data effectively. To achieve improved interpretability for NDE data analysis, this paper proposes a novel dimensionality reduction method, optimized for ultrasonic NDE images, called Gaussian feature approximation (GFA). GFA reduces ultrasonic images to a small number of meaningful descriptors of defect indications, making models trained on these descriptors interpretable and explainable, while still providing accurate defect sizing. GFA operates by fitting a 2D elliptical Gaussian to defect indications in ultrasonic images. Predictions of a ML model trained on GFA features are interpretable because GFA features are based on properties of the defect indication, which are meaningful to a human operator. This contrasts with ML models trained on raw images, where the functional relationship between the input and output is very hard for humans to interpret. Local explanations are enabled by GFA as methods such as SHAP can be used to indicate how individual properties of a defect indication contribute to the defect size prediction. In industrial use this can enable each defect sizing prediction to be presented alongside local feature importance values. These can be viewed by a skilled operator to ensure predictions are being informed by features that are relevant to crack size (e.g., amplitude and indication size in direction of the defect). Global explanations (i.e., describing the average behavior of the sizing model) could also be generated, through the use of techniques such as partial dependence plots and global surrogate models, but these methods are unable to produce the per-sample explanations that SHAP can provide.

To allow comparison of sizing accuracy using GFA, two other well-known methods are applied to create reduced dimensionality feature spaces: principal component analysis (PCA) and the parameters of 6 dB drop boxes fitted around defect indications. Defect sizing is achieved by training a dense neural network on PCA, 6 dB drop and GFA features as well as a convolutional neural network (CNN) [13] on the raw ultrasonic images. CNNs are state of the art for learning from images and using them to address the sizing challenge presented in this paper has been explored previously in [9]. CNNs are used in this paper as a high accuracy, low interpretability, baseline method to compare against.

All three dimensionality reduction methods and their corresponding sizing algorithms are applied to ultrasonic plane wave imaging (PWI) images. Detection is already considered complete, so the target is to size the defects of interest (surface breaking cracks) from the PWI images. The simulation and experimental set-ups are designed to closely approximate the conditions in the example application of this paper: ultrasonic inline pipe inspection. The usefulness of GFA, coupled with a neural network for sizing, and kernel SHAP to produce local explanations, is judged by interpretability, explainability and sizing accuracy. The rest of the paper is structured as follows. Section II describes the inspection setup, and all data sets used in this paper, Section III describes all relevant data pre-processing and analysis methods, Section IV the sizing accuracy and explainability results and Section V conclusions.

II. Inspection Setup and Data Sets

This section describes the inspection setup, the experimental and numerical procedures used to create PWI data, and the parameter space covered by that data. Following on from previous work, these setups and procedures are the same as those used in [9], [14] & [15].

A. Inspection, Imaging and Simulation Methodologies

Inline pipe inspection is a technique often used to assess the integrity of oil and gas pipelines in which a pipeline inspection gauge (PIG) travels within the pipe, making measurements of the surrounding pipe-wall. One of the aims of this inspections is to detects and size surface breaking cracks. These defects can

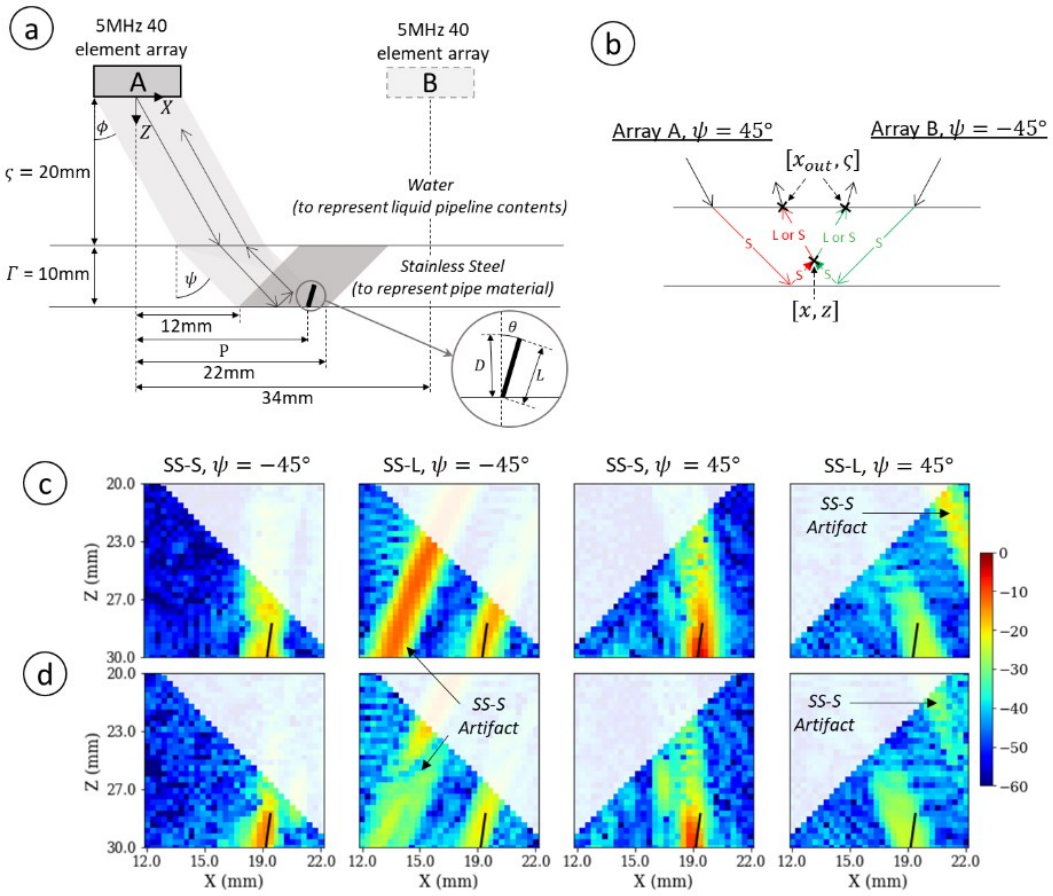

Fig. 1. a) A diagram of the inspection scenario using a plane wave at angle ψ to the sample normal transmitted in the sample with a standoff and thickness of ς and Γ where L, θ and P represent the crack length, angle and position respectively, b) all half-skip shear (S) and longitudinal (L) mode ray-paths used in this paper where x, z are the co-ordinates of the imaging point and x_{out}, ς the co-ordinates of the returning ray on the front wall, c) an example set of simulated images for a defect with $P = 19$ mm, $L = 2$ mm and $\theta = 8°$ and d) a fully experimental set of images for a defect of the same parameters. Note that the black lines show the true extent of the defects, and all images are on the same dB color scale, normalized to the maximum intensity in the experimental set. The areas displayed with more transparency are outside the region insonified by the incident plane wave. Figure reproduced from [9].

occur while the pipeline is in operation, due to environmental factors and in-service stresses. While these cracks can form at any radial location, the stress distribution in pipes means they usually occur on the outer surface. This work focuses on the 'offline' sizing (i.e., once the PIG has been removed from the pipe) of these outer-wall, surface-breaking cracks for oil pipelines, with inline detection assumed already complete. An inspection setup is created to closely approximate in-service conditions without the requirement to access an actual oil pipeline. As illustrated in Fig. 1a a commercially available 5MHz, 0.3 mm pitch, 40 element phased array (Imasonic, Voray-sur-l'Ognon, France) is used to induce shear plane waves in a 10 mm thick stainless-steel plate (approximating a large diameter pipe wall). The array is operated using a Peak NDT (Derby, UK) MicroPulse 5 array controller, receiving on all elements individually, with a sample rate of 50MHz, to form plane wave capture (PWC) data. The array is immersed in water as an approximation for oil that has similar sound speed. Surface-breaking cracks are approximated by 0.3 mm wide electrical discharge machine (EDM) notches.

As shown in Fig. 1b, data is collected from either side of the defect. This is done to replicate the ring of arrays found on a PIG. Each of these arrays fires a normal-incidence wave at $\psi = 0°$ and an angled wave that travels in the fluid at $\phi = \pm 19°$, inducing a $\psi = \pm 45°$ shear wave in the steel plate. The normal-incidence wave is used to calculate standoff (ς) and thickness (Γ), to enable accurate imaging. All sizing is done using the angled waves. An A-Scan is received from the angled wave on each of the array's 40 elements, forming the PWC data. This is then filtered using a Gaussian filter centered at 5 MHz with a -40 dB half width of 4.5 MHz. The filtered PWC data is then focused along a ray path to create images, with the overall process termed Plane Wave Imaging (PWI) [29]. The term ' views' is used to describe ray-paths when more than one is applied to the same physical area. Views are described by the modalities of their transmit and receive legs (L for longitudinal, S for shear) separated by a hyphen. The two views found to be most successful for sizing the surface breaking defects considered in this paper are the SS-S and SS-L half-skip views (i.e., reflecting once off the backwall of the plate in transmission but returning directly in reception). Each array produces an SS-S and SS-L view for each defect, with the region of interest being the full 10 mm depth of plate thickness

and across the insonified region of the backwall, located at 12-22 mm from the array centre in the X-direction. This region is imaged with pixel size (δ) set at the diffraction limit, i.e., half the shear wavelength ($\frac{\lambda_S}{2} = 0.317$ mm), to include all available information without unnecessary computational cost. These two views for the two arrays result in a 32x32x4 set of data.

Simulated data is used to train the neural networks presented in this paper, as creating sufficient experimental training data is prohibitively slow and expensive. To achieve a good trade-off between simulation accuracy and computational expense a hybrid finite element (FE) / ray-based approach is used. The defects are modelled as rectangular, 0.3 mm wide perfect reflectors as there is minimal transmission through them. A local FE simulation is applied to calculate the scattering matrix for defects across all $\{L, \theta\}$ combinations in response to a unimodal plane wave [30]. These scattering matrices are then input into an analytical ray-based model [31], [32] to produce PWC data for all $\{L, \theta, P\}$ combinations. The contributions from grain noise and structural reflections are included by summation with PWC data recorded from a defect-free sample [33]. The resulting A-Scans are then filtered and imaged, as described above, to form the four relevant PWI images. This simulation approach follows the one used in [9], [14], [15].

Example sets of simulated and experimental images for a defect of $P = 19$ mm, $L = 2$ mm and $\theta = 8°$ is given in Fig. 1c. These images are passed through one of the dimensionality reduction methods described in Section III.B before being sized using the neural network described in Section III.C.

TABLE I EXPERIMENTAL DATA SET SUMMARY

The experimental test set contains only the L/θ combinations marked "Test" while the experimental validation set only those marked "Val".

Table reproduced from [15]

		Crack Length, L (mm)				
		1	2	3	4	5
Crack Angle, θ (°)	0	Test	Test	Test	Test	Test
	±2	Test	**Val**	Test	Test	Test
	±5	**Val**	Test	Test	Test	Test
	±8	Test	Test	Test	**Val**	Test
	±15	Test	Test	Test	Test	Test
	±20	Test	Test	**Val**	Test	Test

	Range	*Step*	*Count*
Crack Position, P (mm)	13 to 21	0.3	27

Validation = $N_{\theta,L} \times N_P = 8 \times 27 =$ **216 image sets**

Test = $N_{\theta,L} \times N_P = 47 \times 27 =$ **1269 image sets**

B. Dataset Summary

The application considered in this paper requires the sizing of defects, after their detection. The target is therefore the extent of the defect perpendicular to the surface, $D = L\cos(\theta)$. The parameter space of defects considered is defined by P, L, θ. All experimental defects used are 0.3 mm wide notches on the surface of the stainless-steel plate furthest from the array, manufactured using electrical discharge machining (EDM). Table I summarizes the experimental data. Variation in P is achieved by movement of the array relative to the defect and negative θ achieved by positioning the array on the other side of the defect. This results in a total of 1,485 experimental PWI image sets from the 30 manufactured defects. Table II describes the parameter space coverage of the 16,875 simulated PWI image sets. To enable the ML based sizing algorithms described in Section III.C the simulated and experimental datasets are further split into:

Simulated, Training: 85% (14,343) of simulated data used to optimise the weights and biases of the network.

Simulated, Validation: 7.5% (1,266) of simulated data used during the design stage to qualitatively test for overfitting to the training set.

Simulated, Testing: 7.5% (1,266) of simulated data used to test the sizing accuracy of the network on previously unseen data.

Experimental, Validation: 15% (216) of experimental data used during the design stage to select the network's hyperparameters, test for overfitting to the simulated data and implement the training stop condition.

Experimental, Testing: 85% (1,269) of experimental data used to test the network's sizing accuracy on previously unseen data.

The training/validation/testing split for simulated data is drawn randomly, from a uniform distribution, across all image sets (i.e., across all $\{L, \theta, P\}$). This contrasts with the experimental validation/testing split, which is drawn randomly in $\{L, \theta\}$ space. This distinction guarantees that data from the same physical defect is not split across different sets, ensuring that the L, θ combinations used to demonstrate performance are distinct to the L, θ combinations used to tune the network's hyperparameters and implement the training stop condition.

TABLE II SIMULATED DATA SET SUMMARY

Table reproduced from [15]

Parameter	*Range*	*Step*	*Count*
Crack Length, L (mm)	0.2 to 5	0.2	25
Crack Position, P (mm)	13 to 21	0.3	27
Crack Angle, θ (°)	-24 to 24	2	25
Non-Defect Scan	-	-	36

Total = 25×27×25 = **16,875 image sets**

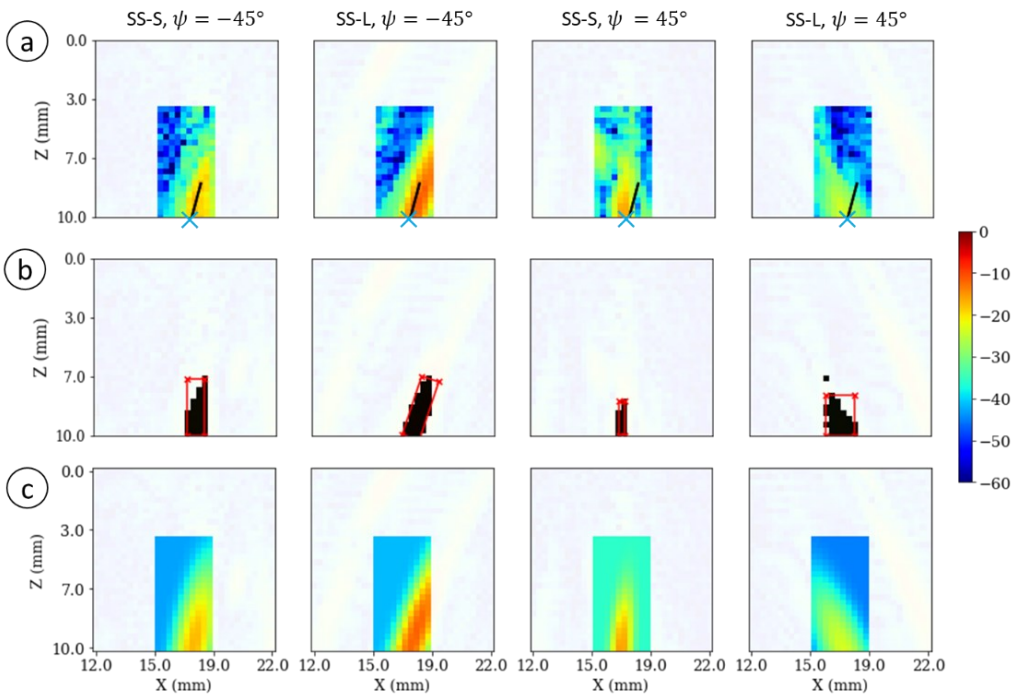

Fig. 2. a) An example of an experimental, windowed PWI image set, from a defect with $P = 17.4\ mm, L = 2\ mm$ and $\theta = 15°$ with the calculated location of the defect on the backwall (x_W) shown as a blue cross and the true extent of the defect shown in black, b) the top 6 dB of the windowed PWI images (in black) and their 6 dB bounding boxes (in red) and c) 2D elliptical Gaussians fit to the windowed PWI images using GFA. In all images the full, unwindowed image, is displayed in the background.

III. Data Processing and Analysis Methods

This section describes the data processing and analysis methods used in this paper: an initial image windowing step, the dimensionality reduction used to improve interpretability, the neural network architectures used to predict defect size, and kernel SHAP: the technique used to produce local explanations.

A. Windowing Images

As exampled in Fig. 1c,d, the PWI image sets in this paper often contain artefacts caused by views other than the one being imaged. This can cause difficulties for sizing algorithms, especially those using transform-coding-based features (such as PCA), as information about the artefact can become 'entangled' with information about the imaged mode. To avoid this, the PWI images are windowed around the defect location before implementing any dimensionality reduction or sizing in this paper. This step is not a fundamental requirement for any of the presented methods, but is a logical pre-processing step, as forcing the model to focus on the location of the defect, and removing unhelpful information, improves sizing accuracy and simplifies explanations. It is also simple to execute for this data set as surface-breaking defects are easy to locate due to their strong corner reflections when insonified at $\psi = \pm 45°$.

Locating a defect on the backwall is implemented by summation of the four associated PWI images (see Fig. 2a for example). The X-location is then found using the maxima in the resulting 32x32 composite image. Using this method on all experimental and simulated data in this paper produces a maximum X-location error of 0.56 mm (1.76 pixels). A window is then applied to the PWI images around the calculated backwall location of the defect (x_W) to isolate the correct indication. In this paper, the window size is set to be 3.15 mm (10 pixels) in X and 6.30 mm (20 pixels) in Z. This window size is selected to be large enough to cover indications from all possible defects within the domain of operation, with minimal contributions from artefacts. An example of a set of windowed PWI images is given in Fig. 2a.

B. Dimensionality Reduction Methods

Three different dimensionality reduction methods are applied to the windowed PWI images in this paper; PCA, 6 dB drop and GFA. The first two of these methods are well-known and presented for comparisons to GFA.

1) Principal Component Analysis

PCA is the process of finding the sequence of orthogonal vectors that best explain the variance of sets of high dimensional data [34]. PCA is often used to find a reduced set of features, with minimal loss of information, for use in ML [35]. In this paper, the principal components are calculated using the windowed, simulated, training set PWI images. The four images per defect are handled individually, using different PCA transforms, to preserve the separation of information between images. M_P different principal components are kept for each 10×20 image. To make the reduced dimensionality consistent with that of with GFA (described in Section 3)), $M_P = 7$ in this paper. As shown in Fig. 3, $M_P = 7$ describes 96% of the variance in the simulated training set, showing that the majority of the information in the images has been captured.

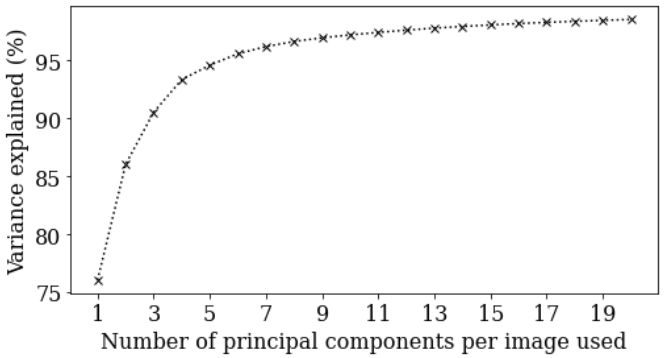

Fig. 3. The variance of the training set captured by different numbers of PCA components.

2) 6 dB Drop

6 dB drop is a well-established defect sizing method in NDE. It is based upon the idea that if a defect is the strongest indicator in an image, the image region within 6 dB of the peak value can be used as a good approximation of the true size of the defect. Traditionally, crack-like defects are sized using the longest edge of a rectangular bounding box that encloses all pixels within 6 dB of the peak [36].

In this paper, the 6 dB drop bounding box is obtained by finding the rectangle with the minimum area that can fit the relevant pixels. The relevant pixels are selected by picking the region of conjoined pixels above -6 dB with the largest total amplitude. Picking the highest conjoined region of high amplitude in this way reduces the chance of noise expanding the size of the box. The SS-L, $\psi = 45°$ image in Fig. 2b shows an example of high amplitude noise excluded by this approach. The 6 dB drop bounding box calculated with this method is used both for dimensionality reduction and as a direct sizing method in this paper. Calculating the parameters of the bounding box (X-position, Z-position, orientation, width and height), results in 5 features per image. These features carry no information directly related to the indication's amplitude. GFA features do contain amplitude information, so for a fair comparison, when sizing from 6 dB box features using the neural networks presented in Section C.2), two additional features are used, resulting in $M_{6dB} = 7$ features. These two additional features are chosen to be the maxima and root mean square (RMS) of all pixels within the bounding box, above -6 dB (i.e., the black pixels in Fig. 2b). Direct, traditional sizing with 6 dB drop is also considered, and is implemented by taking the mean of the longest edges of the boxes fitted to each image.

3) Gaussian Feature Approximation

GFA is a novel dimensionality reduction method presented in this paper with the aim of creating a feature space that is informative (i.e., retains the information needed for accurate defect sizing), interpretable (i.e., meaningful to NDE operators) and improves the quality of local explanations. GFA is performed by fitting a 2D elliptical Gaussian function to each PWI image and using the parameters that define that Gaussian as the features of the image. GFA features describes a defect indication in a similar fashion to 6 dB drop features, but with a more robust fitting procedure that is not dependent on selecting a threshold value, and avoids the need for pre-processing to deal with conjoined pixels. It is also a richer feature space, containing more information about the indications shape, as well as the background noise level. As shown in Section IV.A, these differences make sizing on GFA features significantly more accurate than sizing on 6 dB drop features.

GFA is motivated by the importance of a defect indication's amplitude, spatial size and location in traditional NDE sizing techniques. These underlying features are encoded within PWI images, but not in a form that allows for interpretable models to be trained on them. Fitting an appropriate shape to a PWI image disentangles properties of the defect indication from each other, as well as from information relating to noise and artefacts. The shape used for fitting in GFA is a 2D elliptical Gaussian, this can be described by amplitude at position in the X and Z direction (x, z), given by

$$f_{x,z}(A, x_0, z_0, \sigma_X, \sigma_Z, \Theta, B) = Ae^{-a(x-x_0)^2 - b(x-x_0)(z-z_0) - c(z-z_0)^2} + B \tag{1}$$

$$a = \frac{\cos^2(\Theta)}{2{\sigma_X}^2} + \frac{\sin^2(\Theta)}{2{\sigma_Z}^2},$$

$$b = \frac{\sin(2\Theta)}{2{\sigma_X}^2} - \frac{\sin(2\Theta)}{2{\sigma_Z}^2}, \tag{2}$$

$$c = \frac{\sin^2(\Theta)}{2{\sigma_X}^2} + \frac{\cos^2(\Theta)}{2{\sigma_Z}^2}$$

using seven GFA features: amplitude (A), X-position (x_0), Z-position (z_0), X-sigma (σ_X), Z-sigma (σ_Z), angle (Θ) and offset (B). Finding the optimum set of parameters is achieved by minimizing

$$\ell(A, x_0, z_0, \sigma_X, \sigma_Z, \Theta, B) = \sum_x \sum_z (f_{x,z} - I_{x,z})^2 \tag{3}$$

TABLE III INITIAL GUESS AND BOUNDS FOR GFA FEATURES.
Lower and upper bounds are inclusive.
$\max(I_{x,z})$ *refers to the maxima in the current image for which GFA features are being calculated.*
x_W *is the centre of the 10×20-pixel window and* δ *is the image resolution* ($\delta = \frac{\lambda_S}{2} = 0.317$ mm)
η *is calculated by the root mean square of an experimental PWI image set from a defect free sample.*

Parameter	Amplitude, A	X-position, x_0 (mm)	Z-position, z_0 (mm)	X-sigma, σ_X (mm)	Z-sigma, σ_Z (mm)	Angle, Θ (rad)	Offset, B
Initial guess	$\max(I_{x,z})$	$\text{argmax}(I_{x,z})$	$\text{argmax}(I_{x,z})$	0.5δ	2δ	0	0
Lower bound	0	$x_W - 5\delta$	0	0	0	$-\pi/4$	0
Upper bound	$\max(I_{x,z})$	$x_W + 5\delta$	20δ	10δ	20δ	$\pi/4$	20η

where $I_{x,z}$ is the windowed PWI image and the summations are over the windowed region only. This optimization problem is solved in this paper by using SciPy's curve fitting function [37] with the trust region reflective minimization algorithm [38] as it is particularly suitable for large, bounded problems such as this one. The bounds and initial guess for the seven parameters that define $f_{x,z}$ are described in Table III. It is important to note that bounding x_0 and x_0 within the window is necessary as ℓ has zero gradient when the Gaussian's centre is far away from the window. Also, constraining $-\frac{\pi}{4} < \Theta < \frac{\pi}{4}$ is necessary to ensure there aren't two equivalent solutions with σ_X and σ_Z values swapped.

In principle, more than one Gaussian could be fit to each image. However, for the application presented in this paper, adding a second Gaussian per image and sizing using a neural network (as presented in Section C.2)) was not found to increase sizing accuracy. This is likely because most information useful to the sizing process can be captured by one Gaussian. This is further evidenced by the root mean square error (RMSE) for GFA based sizing only being 23% higher than sizing from the original image (detailed sizing accuracy results are presented in Section IV.A). It should be noted that if fitting more than one Gaussian is deemed necessary it should be done in series (i.e., fit the second Gaussian, $f^2_{x,z}$, to $I_{x,z} - f^1_{x,z}$) rather than in parallel. This is to ensure the ordering of the GFA features is meaningful to the sizing algorithm. More complexity could also be added to $f_{x,z}$ by using more complex-shaped fitting functions with more parameters, such as skewness or properties of background noise, but this would reduce the interpretability of the feature space, so should not be done without certainty that the extra features are informative for the task at hand.

GFA, as introduced in this section, creates a feature space that is implicitly more interpretable than the raw PWI images and enables useful local explanations. GFA features are interpretable as they each uniquely describe a property of the defect indication which is meaningful to an NDE practitioner. They are also only minorly affected by background noise and artefacts, meaning sizing on GFA features is guaranteed to be informed by the defect indication, and not overfitted to other confounding features. Local explanations are made more useful by GFA as they can ascribe importance to specific aspects of a defect indication with GFA features instead of a saliency map in real space. Explainability is further discussed in Sections D and 0.

C. Neural Network Architectures for Defect Sizing

1) Convolutional Neural Network

As a baseline approach with high accuracy and low interpretability the raw PWI images are sized using the CNN designed for this data set in [9]. CNNs are state-of-the-art for image classification tasks due to the power of convolutional layers to map structured, high-dimensional data to informative feature spaces [10]. The CNN architecture used here is illustrated in Fig. 4a. The input is composed of the four 32×32 PWI images stacked in the third dimension, akin to how natural

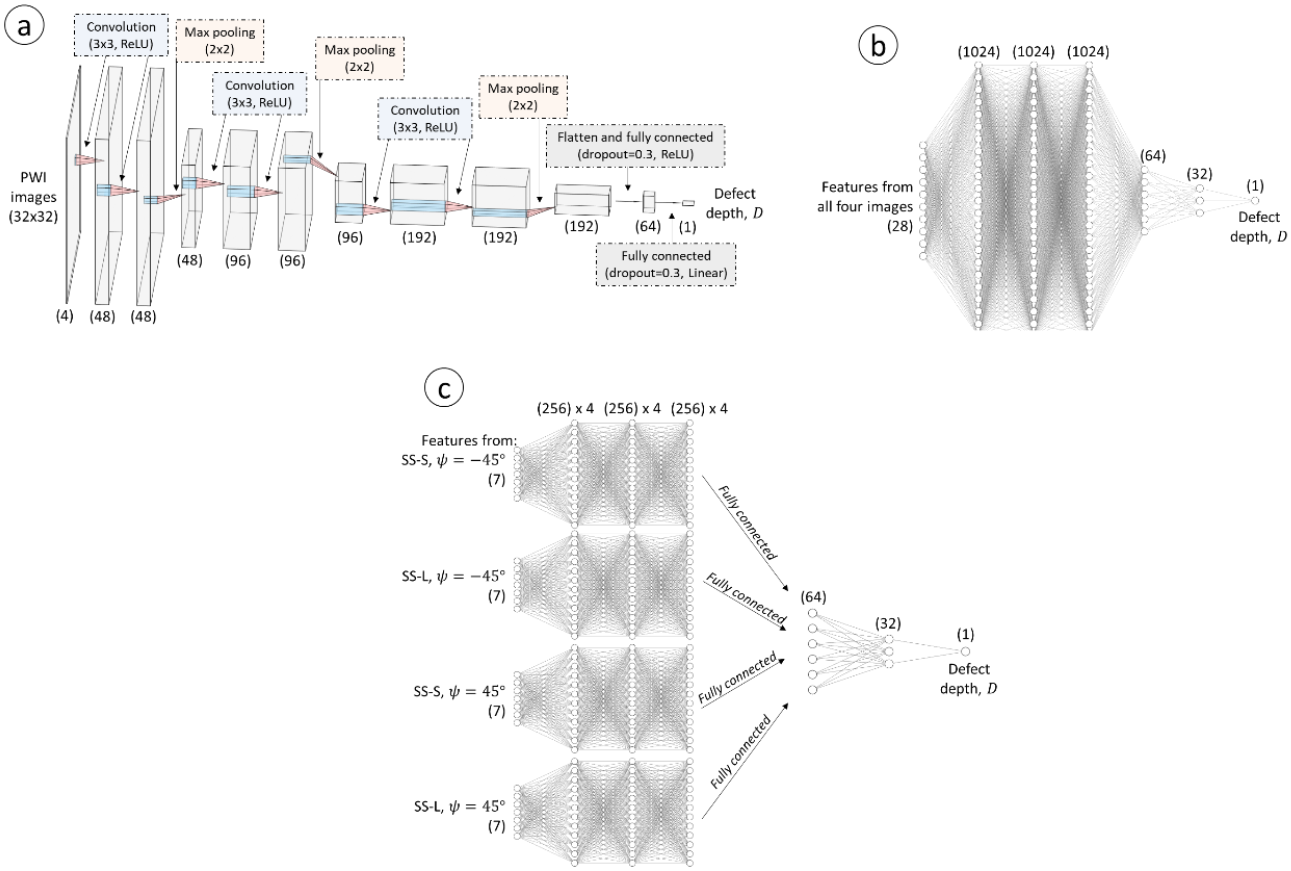

Fig. 4. Neural network architectures used in this paper, as described in Section III.C: a) CNN, b) NN-Single, c) NN-Split.

image CNNs treat red, green and blue channels. The general structure is made up of repeated blocks of convolutional and max-pooling layers for feature extraction, followed by fully connected layers for regression. Rectified linear unit (ReLU) activation is used throughout. Ten percent dropout is applied to the fully connected layer inputs for regularization. The state-of-the-art Adam optimizer [39] is used to train the CNN with a learning rate of 0.001, in mini-batches of 128, with a patience of 150 epochs (i.e., until 150 epochs with no reduction in experimental validation set loss). The network hyperparameters (depth, filter size and number, dropout rate, neuron number etc.) have been selected to optimize experimental validation set accuracy. More details on this design process can be found in [9]. Refer to [40] for ML terminology.

There are three minor changes to the implementation of the CNN between [9] and this paper. Firstly, only a single network is needed to predict D. The network used matches the structure of the L network in [9]. Secondly, dropout is increased to 0.3, which resulted in slightly better experimental validation set accuracy (~4%) at the cost of needing ~50 more epochs to converge. Thirdly, the windowing of the PWI images described in Section A must be accounted for. For computational

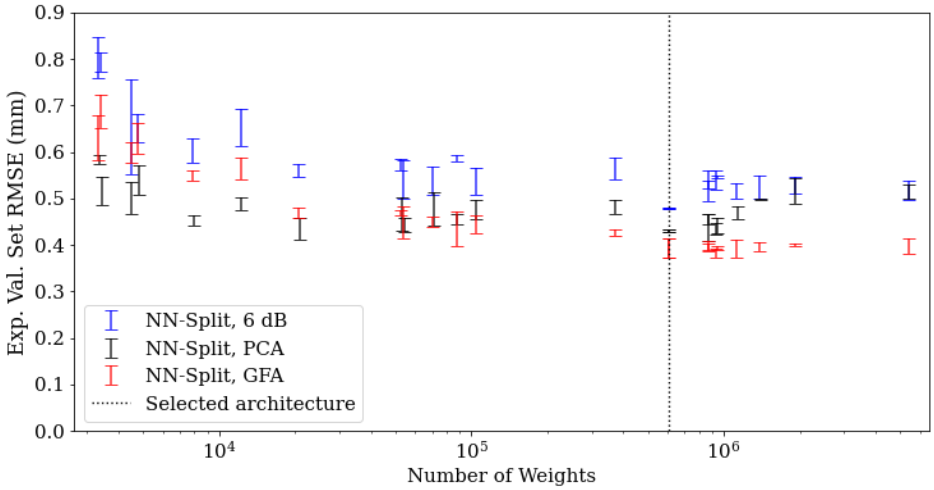

Fig. 5. Experimental validation set RMSE for the NN-Split architecture applied to GFA features, with different hyperparameters. Details of the exact hyperparameters tested are given in Table IV. The error bars represent ± standard deviation over five independent initializations.

TABLE IV HYPERPARAMETERS FOR ALL TESTED NN-SPLIT ARCHITECTURES, AS SHOWN IN FIG. 5.
The selected architecture is highlighted in green.

Number of weights	RMSE (mm)	Neurons in each layer (*before* full connection)	Neurons in each layer (*after* full connection)
3329	0.64	1024, 1024, 1024, 1024	64, 32
3409	0.69	16, 16	64, 32
4481	0.61	16	64, 32
4769	0.62	32, 32	64, 32
7873	0.55	64, 64	64, 32
12225	0.55	64, 64, 64, 64, 64, 64	64, 32
20609	0.47	256	64, 32
52321	0.50	512, 256, 128, 64, 32	64, 32
53889	0.46	256, 256, 256	64, 32
70529	0.45	256, 256, 256	64, 32
87169	0.44	256, 256, 256, 256, 256	64, 32
103809	0.45	256, 256, 256, 256, 256, 256, 0, 0	64, 32
369281	0.42	512, 512, 512, 512, 512, 512	64, 32
602241	0.39	1024, 1024, 1024	64, 32
865409	0.40	1024, 1024, 1024, 1024	64, 32
865921	0.40	1024, 1024, 1024, 1025	64, 32, 16
933057	0.39	1024, 1024, 1024, 1027	128, 32
939265	0.40	1024, 1024, 1024, 1026	128, 64, 32
1128577	0.40	1024, 1024, 1024, 1024, 1024	64, 32
1391745	0.40	1024, 1024, 1024, 1024, 1024, 1024	64, 32
1918081	0.39	1024, 1024, 1024, 1024, 1024, 1024, 1024, 1024	64, 32
5402753	0.39	2048, 2048, 2048, 2048, 2048, 2048	64, 32

efficiency this could, in principle, be done by reducing the input layer size to 10×20×4 and concatenating the X-position with flattened features before the dense layers. However, as the purpose of including CNN-based sizing in this paper is as a baseline for sizing accuracy, computational efficiency is not of major concern. Therefore, the images are simply zero-padded to their original 32×32×4 size before being input into the CNN. This offers a simple way to encode X-position without drastically altering the CNN design and potentially reducing sizing accuracy.

2) Dense Neural Network

Training a sizing algorithm from a set of unstructured numerical features such as those produced by GFA, PCA and 6 dB drop can be done with many ML algorithms (e.g., random forest, support vector machine and k-nearest neighbors). In this paper, sizing from the reduced feature sets is done using a dense neural network, i.e., layers of neurons that are fully connected to preceding layers. This gives a natural comparison with the CNN as both algorithms operate in a similar fashion and have the capability to represent complex, non-linear functions. To match the CNN, the dense neural networks in this paper are trained with the Adam optimizer and use ReLU activation functions on all layers except the input and output. As with CNNs these are also common design choices for dense neural networks. All other hyperparameters are selected via the same design process as presented in [9]; grid search, with selection made using the lowest GFA experimental validation set RMSE. The optimal learning rate was found to be 1×10^{-4}. Application of dropout and L2 regularization were tested but found to increase validation set error, suggesting that they are unnecessary for this reduced dimensionality input data, and so are not used.

In the initial design process for the number of neurons in each layer (i.e., width) and number of layers (i.e., depth), the dense neural network was set to follow a common structure: a sequential set of fully connected layers of reducing width. This architecture is illustrated in Fig. 4b and referred to as NN-Single from here onwards. However, in following iterative design stages it was found that fixing the number of neurons but removing connections between the features from different image modes improved performance. This produces a structure of four dense neural networks, fully connected in the final few layers. This architecture is illustrated in Fig. 4c and referred to as NN-Split from here onwards. The authors believe that NN-Split outperforms NN-Single, even with the same number of neurons, because it allows the initial layers to compose the features from an individual image into a more expressive form without immediately entangling them with features from other images. The experimental validation set RMSE for all NN-Split widths and depths tested are given in Fig. 5, and their hyperparameters described in Table IV. As found in [9] for the design of the CNN, NN-Split with GFA and 6 dB features shows a 'diminishing returns' relationship between the number of weights (here used as a proxy for complexity) and sizing accuracy. PCA features provide good sizing accuracy even with the lowest complexity networks tested. The architecture selected (indicated by a dashed line in Fig. 5, and illustrated in Fig. 4c) is deemed to be a good trade-off between computational complexity and performance for GFA features, and provides good sizing accuracy with all three feature types. It is interesting to note that despite the input data dimensionality reduction of 96.5% (i.e., from $10 \times 20 \times 4$ to 7×4) the number of weights in NN-Split are only 76% lower than in CNN. This suggests that the relationship between both 6 dB drop and GFA features, and crack size is still very complex and non-linear, despite the dimensionality reduction. Understanding why PCA features require a significantly less complex neural network to achieve good sizing accuracy requires further research.

D. Local Explanations using Kernel SHAP

There are many popular methods for creating local explanations for ML predictions. A unifying view for these methods, termed SHAP, has been presented in [22]. SHAP aims to produce game theory results (i.e., Shapley values [41], [42]) in a computationally efficient manner and unifies most modern model explanation methods (LIME [18], DeepLIFT [19], [20], LRP [21] and classic Shapley estimation methods [43]–[45]) as different versions of the same framework. The underlying logic behind SHAP is to approximate the output of the original prediction model, given the current input, $f(x)$, with a linear explanation model, given a set of simplified inputs (e.g., bag of words for text features or saliency maps for images),

$$f(x) \approx g(z') = \varphi_0 + \sum_{i=1}^{M} \varphi_i z_i' \tag{4}$$

where $z' \in \{0,1\}^M$, M is the number of simplified input features and φ_i the importance of each feature (i.e., the SHAP values). φ_0 is set to be the mean of each feature in the model's training set in this paper, as is common in most published implementations. φ_i is a function of the current input, x.

If features are assumed to be independent when approximating conditional expectations, as in LIME and

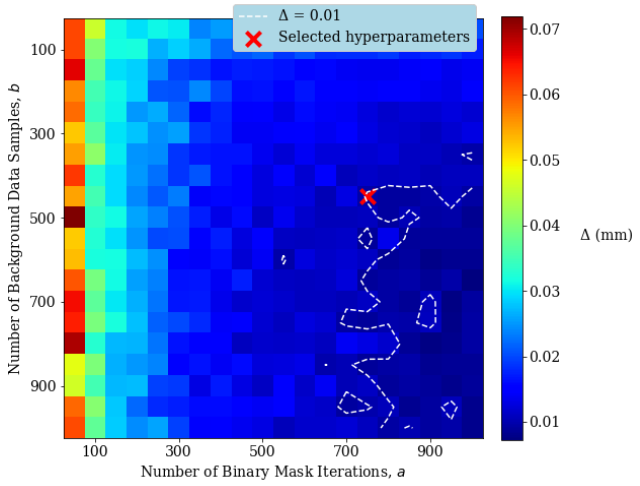

Fig. 6. The change in mean absolute SHAP values ($\Delta = \sum_{i=0}^{5}\sum_{j=0}^{28}|\varphi_{i,j,a,b} - \varphi_{i,j,5000,5000}|$) for NN-Split, sizing from GFA features, for different numbers of binary mask iteration (a) and background data samples (b). $\Delta = 0.01$ is deemed to be sufficiently low so a contour at this level (as described by the white dotted line) is used to select a and b.

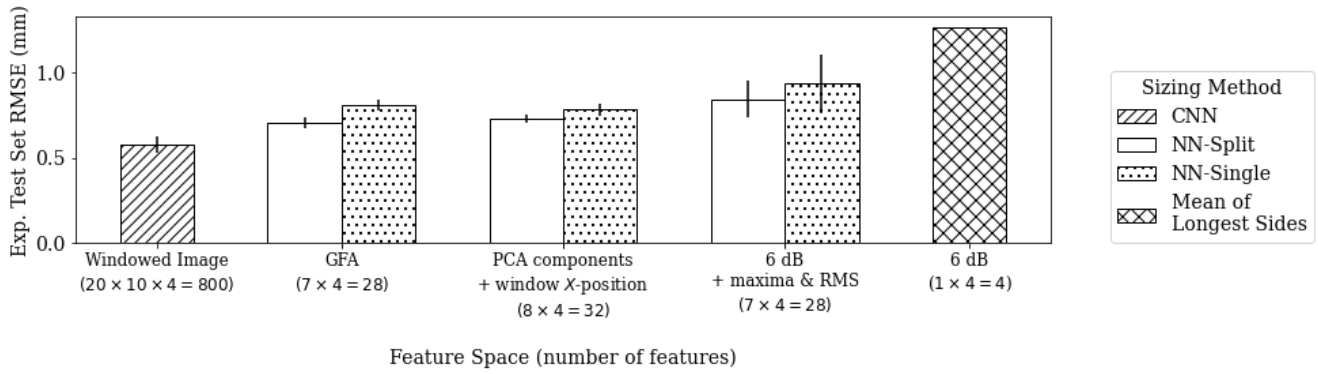

Fig. 7. RMSE across the full experimental test set (1269 sets of PWI images) for all dimensionality reduction methods and associated sizing methods discussed in this paper. The error bars represent over thirty independent initializations.

DeepLIFT, then SHAP values can be estimated directly using the Shapley sampling values method [45]. This involves uniformly sampling permutations of z_i' . Note that for most applications, setting a feature to 0 does not effectively represent the absence of that feature ($z_i' = 0$), so instead, that feature is set to a value sampled from the training set. The issue with the Shapley sampling values method is that sampling enough to get an accurate explanation is slow to compute for large numbers of inputs. Kernel SHAP [22] is a more computationally efficient sampling method as it jointly estimates all φ_i using a linear regression formulation, leading to fewer required samples for accurate estimation of Shapley values. This is achieved by weighting samples of z' by a kernel,

$$k(z') = \frac{M-1}{\binom{M}{s} s(M-s)} \tag{5}$$

where s is the number of ones in z' and $\binom{M}{s}$ represents M choose s. This is a very similar approach to that of LIME, but removes the need to select a loss function, weighting kernel, or regularizer, while guaranteeing local accuracy, missingness and consistency (as defined in [22]) in the explanation. The only two hyperparameters for kernel SHAP are the number of binary mask iterations (a) and samples of training data in the background data set (b). The number of iterations to calculate SHAP values is $a \times b$.

To select an a and b that ensures sufficient sampling, without excessive computation, a grid search is carried out. For each combination of a and b tested the SHAP values of five random experimental test set image sets, for a NN-Split model using GFA features, are calculated. The mean absolute difference between these SHAP values and those calculated with a large number of samples ($a = b = 5000$) is calculated,

$$\Delta = \sum_{i=0}^{5} \sum_{j=0}^{28} \left| \varphi_{i,j,a,b} - \varphi_{i,j,5000,5000} \right| \tag{6}$$

where $\varphi_{i,j,a,b}$ represents the SHAP value for data set i and feature j. The results of this for grid search are displayed in Fig. 6. $\Delta = 0.01$ is considered to indicate sufficient convergence, so $a = 750$, $b = 450$ are selected for use in the rest of this paper as this is the minimum number of kernel SHAP samples necessary to achieve $\Delta = 0.01$.

IV. Results

This section gives a comparison of sizing accuracy when using the dimensionality reduction techniques presented in III.B with the ML architectures presented in III.C. The interpretability of the presented sizing networks is discussed and local explanations of predictions using GFA with NN-Split are presented.

A. Sizing Accuracy

While sizing accuracy is not the main focus of this paper, an interpretable defect sizing algorithm that can't size reasonably accurately is not of any use. The current gold-standard accuracy for this data set is CNN-based sizing on the raw ultrasonic images [9]. As shown in Fig. 7, this provides a RMSE of 0.58 mm. Note that this is 29% lower than the same architecture trained on unwindowed images, proving the value of removing information unrelated to the task at hand. For all the ML based sizing methods thirty independently trained networks are trained, with the bars in Fig. 7 displaying the mean RMSE and ± one standard deviation plotted as error bars.

The least accurate sizing is provided by 6 dB drop. Both training a NN-Split on the 7 × 4 parameters of the 6 dB drop box (including maxima and RMS amplitude information) and directly using the mean of the longest sides (as is the traditional method) for sizing produces poor sizing with an experimental test set RMSE of 0.843 mm and 1.26 mm respectively. Note that just using the longest edge of one SS-S image gives 33% higher RMSE than the mean of all four. The high sizing error when using 6 dB drop features is likely because they do not carry enough information relevant to sizing the defects. The next most accurate sizing technique is NN-Split, using 7 PCA components, concatenated with the window's X-position. This gives a RMSE of 0.73 mm. GFA with NN-Split offers the closest sizing accuracy to the CNN with a RMSE of 0.71 mm, despite having only 7 features per image. NN-Single predictions on GFA data are 14% higher than NN-Split predictions on the same data. This motivates the use of NN-Split as it offers better sizing accuracy despite containing 73% less weights.

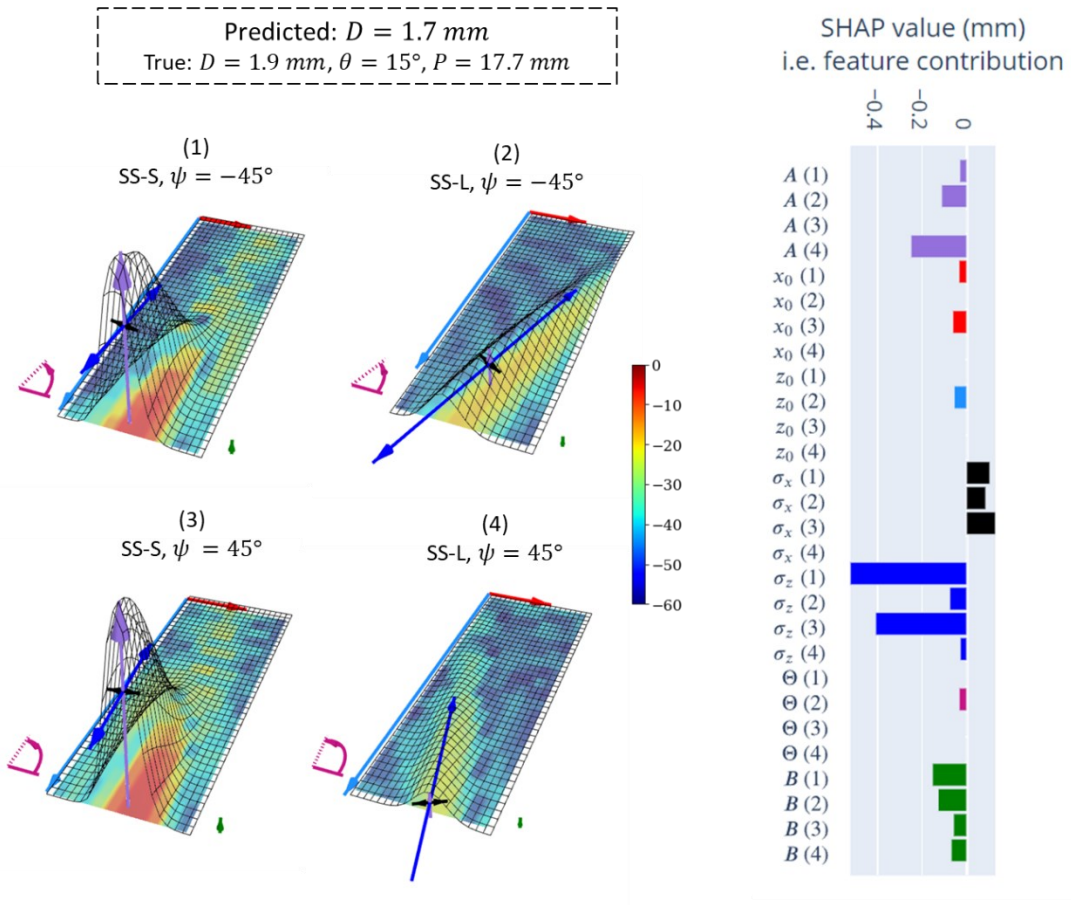

Fig. 8. An example explanation visualization for a sizing prediction from an experimental test set example with $P = 17.7\ mm, L = 2\ mm$ and $\theta = 15°$. The sizing is achieved using NN-Split and GFA features and the feature contributions calculated using kernel SHAP. The 3D plots display the original PWI image as a colourmap, a visualization of the GFA fit as a wireframe surface and the GFA features drawn with arrows colored in relation to the SHAP bar chart. An interactive version of this figure can be found at *https://richardp1234.github.io/GFA-Vis/index4.html*.

B. Interpretability and Explainability

As discussed in Section I, training a CNN on raw ultrasonic images is a 'black box' approach, as it is not interpretable, and local explanations are limited to saliency maps. PCA provides a lower dimensionality input data but neither the magnitude nor form of the components has physical meaning, so are not explainable or interpretable. Sizing based on GFA and 6 dB drop box features is interpretable as the features are simple descriptors of the defect indication. Also, as these dimensionality reduction methods are only minorly affected by background noise and artefacts, the operator can be confident that sizing predictions are informed only by the defect indication. Despite the similar levels of interpretability, GFA is significantly more useful than 6 dB drop due to the significantly lower sizing error, as presented in Section A.

As well as the implicit interpretability provided by sizing with GFA features, useful local explanations can be created with them. As described in Section III.D, SHAP values can be calculated to indicate the importance of each feature to the sizing prediction for a specific defect. An example of how this could be visualized for an operator is given in Fig. 8. The magnitude of the SHAP values indicate how important each feature is to the prediction and their sign (i.e., positive or negative) shows whether that feature is pushing the prediction higher or lower from φ_0. For the example in Fig. 8, the most impactful features are the σ_Z of the SS-S views. This makes intuitive sense as all defects in this paper are oriented roughly in the Z-direction and in these two views there is high amplitude specular reflections from the full extent of the defect. Visualizing local explanations of interpretable ML models in this way can allow operators to interrogate individual predictions, building trust in them and spotting occasions when they are not functioning as expected.

To analyze overall feature importance for the trained NN-Split model, SHAP values are calculated for every defect in the experimental test set. These SHAP values are then visualized in a 'bee-swarm' plot [46] in Fig. 9. In this plot each defect is represented as 28 dots, one for each feature. A dot's color represents the normalized magnitude of the feature and its X-position is determined by SHAP value. Dots placed at the same X-position are plotted with different vertical positions to avoid covering each other. The features are sorted by their mean absolute SHAP value

$$F_j = \frac{1}{N}\sum_{i=0}^{N}\left|\varphi_{i,j}\right| \tag{7}$$

where i represents the index of the sample in the test set and j

the feature. Viewing explanations for the entire test set in this manner is a useful tool for developing ML algorithms as it allows the engineer to ensure the features expected to be informative are consistently given the most importance.

F for the top four features in Fig. 9 is significantly larger than for others, indicating that these are the features that impact crack size prediction the most. The fact that σ_Z for the two SS-S views is considered important by the network builds trust in its predictions. This is because the SS-S view produces the indications most closely matching the true extent of the defect and σ_Z is similar to the parameter used by traditional 6 dB sizing. The fact that the SS-L view amplitudes are the top two most impactful features is an interesting result. It is the authors' belief that this may be where the network is inferring θ, to account for the variations in amplitude and σ_Z it causes. This hypothesis is based on the high correlation between true crack angle and the angle of the indication (i.e., θ and Θ) in these views (Spearman's correlation = 0.56). Fig. 9 also shows that the features not used by classical NDE sizing techniques are assigned low F. These include, for example, the width of the indication (σ_X) and the background noise level (B).

The correlations between the value of features (color of the dots in Fig. 9) and their impact on network output (X-position of the dots in Fig. 9) can also be in inspected, and here too we find similar behaviors as found in classic physics-based NDE sizing methods:

- A and σ_Z values are positively correlated with defect size.
- The nearer a defect is to the array (low x_0 for $\psi = 45°$, high x_0 for $\psi = -45°$), the larger the SHAP value. This positively contributes to the D prediction, correcting for the fact that defects near to the array appear smaller, because they are not insonified over their full extent.
- Defects angled towards the array (-ve Θ for $\psi = 45°$, +ve Θ for $\psi = -45$) have larger SHAP values. This positively contributes towards the D prediction, correcting for the fact that these indications have significantly lower amplitude in SS-L modes.

There are more complex interactions happening with z_0, σ_x and B that are harder to draw conclusions from in Fig. 9, but they are also lower in average feature importance (F) and have less significance for physics-based defect sizing.

There are a few outliers in Fig. 9. Most notably, there are three samples with significantly larger |SHAP| values for σ_X in the SS-L views (i.e. σ_X (2) and σ_X (4)) when compared to the rest of the test set. These three sample are also outliers in terms of their σ_X values as GFA has fit a wide Gaussian to the image. This fitting has occurred because there is only a very low amplitude response from the defect in these views, leading GFA to mostly fit to variations in the background noise. It is hypothesized that this raised feature importance is an indication of the sizing network using this increased σ_X as a way to detect a view with little to no signal in, however, analyzing outliers in this fashion is challenging. Unexpected SHAP value outcomes such as these indicate sizing predictions that should be flagged for further analysis by a human and/or more advanced automated analysis. This highlights, again, the usefulness of local explainability.

V. Conclusions

This paper has presented GFA, a novel dimensionality reduction method, aimed at improving the interpretability and explainability of ML for ultrasonic NDE. Defect sizing with a neural network (NN-Split), trained on simulated GFA features, tested on experimental data, has been shown to produce a RMSE only 23% higher than a CNN applied to the full PWI image sets, despite the dimensionality reduction from $10 \times 20 \times 4$ to 7×4. The other dimensionality reduction methods tested all provided comparable or worse sizing performance. In terms of interpretability, GFA improves upon both the original images and PCA. 6 dB drop features have comparable interpretability to GFA but provide significantly less accurate sizing.

GFA provides improved interpretability to models that use the features as, unlike individual pixel values, their values are meaningful to NDE operators. Also, as GFA features are only minorly affected by artefacts and background noise, sizing on GFA features is guaranteed to be informed by the defect indication, and not overfitted to other confounding features.

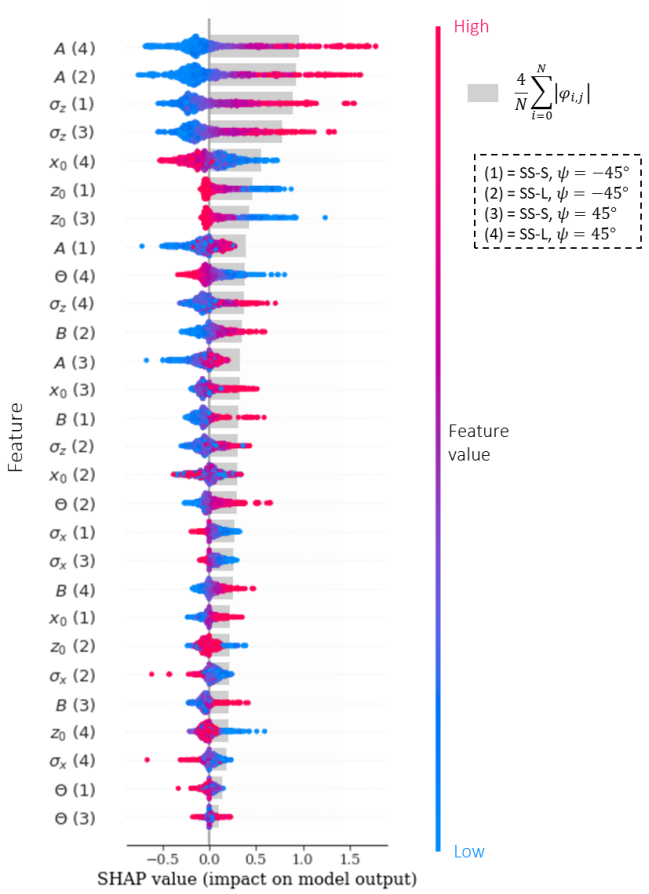

Fig. 9. A bee-swarm plot of the SHAP values for the experimental test set with sizing achieved using NN-Split and GFA features. (n) represents the n_{th} imaging mode as indicated in Fig. 8. The features are sorted by $F_j = \frac{1}{N}\sum_{i=0}^{N}|\varphi_{i,j}|$ where i represents the index of the sample in the test set and j the feature. $4 \times F$ is plotted as gray bars, this has been scaled by 4 in this plot for ease of visual comparison between features.

GFA also enables useful local explanations as methods such as kernel SHAP can be used to inform the operator which features are important to the sizing of a specific defect and if that feature contributes positively or negatively to the prediction. It should be noted that SHAP values are a simplification of the model, assuming a linear combination of independent features, so cannot fully explain global model behavior. However, they can still be of great use in building trust in a model's predictions, by comparison with the decision-making process of expert intuition or physics-based sizing approaches.

If further interpretability was desired beyond that of GFA, as described in this paper, the feature with the lowest average SHAP value (i.e., lowest F_j) could be removed from the training set and the network retrained using only the remaining features. This could be done iteratively until validation set RMSE became unacceptably large, or the feature space was deemed to be small enough to have satisfactory interpretability. This iterative training approach could also be used in applications where it is useful to discover which GFA feature is the most impactful to the task at hand, akin to the aim of sparse identification of nonlinear dynamics (SINDy) [47].

GFA, as presented in this paper, is readily applicable to all ultrasonic NDE image analysis. If windowing around one defect indication per image is not possible, the iterative fitting of more Gaussians can be used to better capture the useful information. In general, fitting functions to NDE data to reduce its dimensionality is an approach that is generalizable to other modalities and data structures (e.g., electromagnetic NDE data and ultrasonic B-Scans) and increasingly, a computationally tractable task. 2D elliptical Gaussians, as used in GFA, are effective for ultrasonic images of approximately straight crack-like defects, but using a different function will be necessary when defect indications are a significantly different shape. For example, parameterizing the curvature of a defect indication would be necessary for analyzing B-Scan data or more complex defects.

Appendix

Supporting code and data are available at the University of Bristol data repository, data.bris, at https://data.bris.ac.uk/data/dataset/2o82rzo6d5ly32h7msblzq4y8v.

References

[1] L. Udpa and S. S. Udpa, "Neural networks for the classification of nondestructive evaluation signals," *IEE Proceedings, Part F: Radar and Signal Processing*, vol. 138, no. 1, pp. 41–45, 1991, doi: 10.1049/ip-f-2.1991.0007.

[2] N. Amiri, G. H. Farrahi, K. R. Kashyzadeh, and M. Chizari, "Applications of ultrasonic testing and machine learning methods to predict the static & fatigue behavior of spot-welded joints," *J Manuf Process*, vol. 52, pp. 26–34, Apr. 2020, doi: 10.1016/j.jmapro.2020.01.047.

[3] M. Mishra, A. S. Bhatia, and D. Maity, "Predicting the compressive strength of unreinforced brick masonry using machine learning techniques validated on a case study of a museum through nondestructive testing," *J Civ Struct Health Monit*, pp. 1–15, Mar. 2020, doi: 10.1007/s13349-020-00391-7.

[4] Z. Lin, H. Pan, G. Gui, and C. Yan, "Data-driven structural diagnosis and conditional assessment: from shallow to deep learning," in *Sensors and Smart Structures Technologies for Civil, Mechanical, and Aerospace Systems 2018*, Mar. 2018, vol. 10598, p. 38. doi: 10.1117/12.2296964.

[5] J. Ye, S. Ito, and N. Toyama, "Computerized ultrasonic imaging inspection: From shallow to deep learning," *Sensors (Switzerland)*, vol. 18, no. 11, Nov. 2018, doi: 10.3390/s18113820.

[6] I. Virkkunen, T. Koskinen, O. Jessen-Juhler, and J. Rinta-Aho, "Augmented ultrasonic data for machine learning," *J Nondestr Eval*, vol. 40, no. 1, pp. 1–11, 2021.

[7] S. Sambath, P. Nagaraj, and N. Selvakumar, "Automatic defect classification in ultrasonic NDT using artificial intelligence," *J Nondestr Eval*, vol. 30, no. 1, pp. 20–28, Mar. 2011, doi: 10.1007/s10921-010-0086-0.

[8] X. L. Travassos, S. L. Avila, and N. Ida, "Artificial neural networks and machine learning techniques applied to ground penetrating radar: A review," *Applied Computing and Informatics*, 2020.

[9] R. J. Pyle, R. L. T. Bevan, R. R. Hughes, R. K. Rachev, A. Ait Si Ali, and P. D. Wilcox, "Deep Learning for Ultrasonic Crack Characterization in NDE," *IEEE Trans Ultrason Ferroelectr Freq Control*, vol. 68, no. 5, pp. 1854–1865, 2020, doi: 10.1109/TUFFC.2020.3045847.

[10] A. Voulodimos, N. Doulamis, A. Doulamis, and E. Protopapadakis, "Deep learning for computer vision: A brief review," *Comput Intell Neurosci*, vol. 2018, 2018.

[11] A. S. Lundervold and A. Lundervold, "An overview of deep learning in medical imaging focusing on MRI," *Z Med Phys*, vol. 29, no. 2, pp. 102–127, 2019.

[12] K. Suzuki, "Overview of deep learning in medical imaging," *Radiological Physics and Technology*, vol. 10, no. 3. Springer Tokyo, pp. 257–273, Sep. 01, 2017. doi: 10.1007/s12194-017-0406-5.

[13] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[14] R. J. Pyle, R. L. T. Bevan, R. R. Hughes, A. A. S. Ali, and P. D. Wilcox, "Domain Adapted Deep-Learning for Improved Ultrasonic Crack Characterization Using Limited Experimental Data," *IEEE Trans Ultrason Ferroelectr Freq Control*, vol. 69, no. 4, pp. 1485–1496, 2022, doi: 10.1109/TUFFC.2022.3151397.

[15] R. J. Pyle, R. R. Hughes, A. A. S. Ali, and P. D. Wilcox, "Uncertainty Quantification for Deep Learning in Ultrasonic Crack Characterization," *IEEE Trans Ultrason Ferroelectr Freq Control*, vol. 69, no. 7, pp. 2339–2351, 2022, doi: 10.1109/TUFFC.2022.3176926.

[16] C. Rudin, "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead," *Nat Mach Intell*, vol. 1, no. 5, pp. 206–215, 2019.

[17] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 618–626.

[18] M. T. Ribeiro, S. Singh, and C. Guestrin, "' Why should i trust you?' Explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 2016, pp. 1135–1144.

[19] A. Shrikumar, P. Greenside, A. Shcherbina, and A. Kundaje, "Not just a black box: Learning important features through propagating activation differences," *arXiv preprint arXiv:1605.01713*, 2016.

[20] A. Shrikumar, P. Greenside, and A. Kundaje, "Learning important features through propagating activation differences," in *International conference on machine learning*, 2017, pp. 3145–3153.

[21] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller, and W. Samek, "On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation," *PLoS One*, vol. 10, no. 7, p. e0130140, 2015.

[22] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," *Adv Neural Inf Process Syst*, vol. 30, 2017.

[23] L. H. Gilpin, D. Bau, B. Z. Yuan, A. Bajwa, M. Specter, and L. Kagal, "Explaining explanations: An overview of interpretability of machine learning," in *2018 IEEE 5th International Conference on data science and advanced analytics (DSAA)*, 2018, pp. 80–89.

[24] A. Karthikeyan, A. Tiwari, Y. Zhong, and S. T. S. Bukkapatnam, "Explainable AI-infused ultrasonic inspection for internal defect detection," *CIRP Annals*, 2022.

[25] L. Fradkin, S. Uskuplu Altinbasak, and M. Darmon, "Towards Explainable Augmented Intelligence (AI) for Crack Characterization," *Applied Sciences*, vol. 11, no. 22, p. 10867, 2021.

[26] H. Zhang, J. Lin, J. Hua, and T. Tong, "Interpretable convolutional sparse coding method of Lamb waves for damage identification and localization," *Struct Health Monit*, p. 14759217211044806, 2021.

[27] C. Schnur *et al.*, "Towards interpretable machine learning for automated damage detection based on ultrasonic guided waves," *Sensors*, vol. 22, no. 1, p. 406, 2022.

[28] A. Zytek, I. Arnaldo, D. Liu, L. Berti-Equille, and K. Veeramachaneni, "The Need for Interpretable Features," *ACM SIGKDD Explorations Newsletter*, vol. 24, no. 1, pp. 1–13, 2022, doi: 10.1145/3544903.3544905.

[29] L. le Jeune, S. Robert, E. L. Villaverde, and C. Prada, "Plane Wave Imaging for ultrasonic non-destructive testing: Generalization to multimodal imaging," *Ultrasonics*, vol. 64, pp. 128–138, 2016.

[30] P. D. Wilcox and A. Velichko, "Efficient frequency-domain finite element modeling of two-dimensional elastodynamic scattering," *J Acoust Soc Am*, vol. 127, no. 1, pp. 155–165, Jan. 2010, doi: 10.1121/1.3270390.

[31] R. K. Rachev, P. D. Wilcox, A. Velichko, and K. L. McAughey, "Plane Wave Imaging Techniques for Immersion Testing of Components with Non-Planar Surfaces," *IEEE Trans Ultrason Ferroelectr Freq Control*, pp. 1–1, Jan. 2020, doi: 10.1109/tuffc.2020.2969083.

[32] L. W. Schmerr, *Fundamentals of ultrasonic nondestructive evaluation*. Springer, 2016.

[33] H. A. Bloxham, A. Velichko, and P. D. Wilcox, "Combining simulated and experimental data to simulate ultrasonic array data from defects in materials with high structural noise," *IEEE Trans Ultrason Ferroelectr Freq Control*, vol. 63, no. 12, pp. 2198–2206, 2016.

[34] K. Pearson, "LIII. On lines and planes of closest fit to systems of points in space," *The London, Edinburgh, and Dublin philosophical magazine and journal of science*, vol. 2, no. 11, pp. 559–572, 1901.

[35] F. Kherif and A. Latypova, "Principal component analysis," in *Machine Learning*, Elsevier, 2020, pp. 209–225.

[36] J. Zhang, B. W. Drinkwater, and P. D. Wilcox, "The use of ultrasonic arrays to characterize crack-like defects," *J Nondestr Eval*, vol. 29, no. 4, pp. 222–232, 2010.

[37] SciPy, "scipy.optimize.curve_fit," 2022. https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.least_squares.html (accessed Jul. 12, 2022).

[38] M. A. Branch, T. F. Coleman, and Y. Li, "A subspace, interior, and conjugate gradient method for large-scale bound-constrained minimization problems," *SIAM Journal on Scientific Computing*, vol. 21, no. 1, pp. 1–23, 1999.

[39] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, Dec. 2015.

[40] Google, "Machine Learning Glossary | Google Developers," *Machine Learning Crash Course*. https://developers.google.com/machine-learning/glossary (accessed Jun. 10, 2020).

[41] L. Shapley, "Quota solutions op n-person games1," *Edited by Emil Artin and Marston Morse*, p. 343, 1953.

[42] A. E. Roth, *The Shapley value: essays in honor of Lloyd S. Shapley*. Cambridge University Press, 1988.

[43] A. Datta, S. Sen, and Y. Zick, "Algorithmic transparency via quantitative input influence: Theory and experiments with learning systems," in *2016*

IEEE symposium on security and privacy (SP), 2016, pp. 598–617.

[44] S. Lipovetsky and M. Conklin, "Analysis of regression in game theory approach," *Appl Stoch Models Bus Ind*, vol. 17, no. 4, pp. 319–330, 2001.

[45] E. Štrumbelj and I. Kononenko, "Explaining prediction models and individual predictions with feature contributions," *Knowl Inf Syst*, vol. 41, no. 3, pp. 647–665, 2014.

[46] S. Lundberg, "beeswarm plot," 2018. https://shap.readthedocs.io/en/latest/example_notebooks/api_examples/plots/beeswarm.html (accessed Aug. 12, 2022).

[47] S. L. Brunton, J. L. Proctor, and J. N. Kutz, "Discovering governing equations from data by sparse identification of nonlinear dynamical systems," *Proceedings of the national academy of sciences*, vol. 113, no. 15, pp. 3932–3937, 2016.

Richard J. Pyle was born in Torquay, U.K. in 1996. He received an M.Eng. degree in mechanical engineering from The University of Bristol, U.K. in 2018.

Through the summer of 2017 he worked for Cavendish Nuclear as a graduate design engineer. He is now studying for an Eng.D. degree in ultrasonic phased array signal processing at The University of Bristol, sponsored by Baker Hughes, Cramlington, U.K. His current research interests include phased array imaging, data compression, defect characterization and machine learning.

Robert R. Hughes was born in Bristol, U.K., in 1989. He received an M.Phys. degree in physics followed by an Engineering Doctorate (Eng.D.) in non-destructive evaluation from the Department of Physics, University of Warwick, in 2016. His Eng.D. research was sponsored by Rolls-Royce plc., Bristol, where he carried out an industrial placement between 2014 and 2015 focusing on eddy-current array sensor development and data-analysis.

In 2015, Dr. Hughes took up a position as Research Associate with the Department of Mechanical Engineering, University of Bristol, U.K, where he developed eddy-current inspection and data-analysis techniques for characterising surface-breaking defects and carbon-fibre composite structures. From 2019, Dr. Hughes has been a Lecturer in non-destructive testing at the Department of Mechanical Engineering, University of Bristol, U.K where his current research interests include eddy-current inspection, inversion of inhomogenous materials, defect characterisation and advanced data-analysis techniques, as well as magnetic particle sensing & manipulation in microfluidic environments.

Paul D. Wilcox was born in Nottingham (England) in 1971. He received an M.Eng. degree in Engineering Science from the University of Oxford (Oxford, England) in 1994 and a Ph.D. from Imperial College (London, England) in 1998. He remained in the Non-Destructive Testing (NDT) research group at Imperial College as a Research Associate until 2002, working on the development of guided wave array transducers for large area inspection.

Since 2002 Prof. Wilcox has been with the Department of Mechanical Engineering at the University of Bristol (Bristol, England) where his current title is Professor of Dynamics. He held an EPSRC Advanced Research Fellowship in Quantitative Structural Health Monitoring from 2007 to 2012, was Head of the Mechanical Engineering Department from 2015 to 2018, and has been a Fellow of the Alan Turing Institute for Data Science since 2018. In 2015 he was a co-founder of Inductosense Ltd., a spin-out company which is commercialising inductively-coupled embedded ultrasonic sensors. His research interests include array transducers, embedded sensors, ultrasonic particle manipulation, long-range guided wave inspection, structural health monitoring, elastodynamic scattering and signal processing.