Towards a multi-fidelity deep learning framework for a fast and realistic generation of ultrasonic multi-modal Total Focusing Method images in complex geometries

Gerardo Granados, Roberto Miorelli, Filippo Gatti, Sébastien Robert, Didier Clouteau

Towards a multi-fidelity deep learning framework for a fast and realistic generation of ultrasonic images in complex geometries with the multi-modal Total Focusing Method

GE Granados[1], R Miorelli[1], F Gatti[2], S Robert[1], D Clouteau[2]
[1]Université Paris-Saclay, CEA, List, F-91120 Palaiseau, France
[2]Laboratoire de Mécanique Paris-Saclay, CentraleSupélec, ENS Paris-Saclay, CNRS, Gif-sur-Yvette, France

31 december 2022

Abstract

This paper presents a deep-learning surrogate model tailored for a fast generation of realistic ultrasonic images in the Multi-modal Total Focusing Method (M-TFM) framework. The method employs both physics- and data- driven data-sets. To this end, we propose a Conditional U-Net (cU-Net) to perform a controlled generative process of high-resolution M-TFM images by spanning the set of inspection parameters, employing both the experimental data (high-fidelity acquisitions) and the simulated ones (a low-fidelity counterpart). Once trained on experimental and simulated images, the cU-Net embodies an enhanced realism, learnt from the experimental data, coupled with a quasi-real-time prediction that prevents the need for extra simulations. Moreover, our surrogate model provides a controlled M-TFM generation conditioned by the steering parameters of the simulation as well as by the physics underlying the ultrasonic testing schema. The performances of our approach are demonstrated in a case study of M-TFM images of a component with planar defects in a complex weld-like profile. Furthermore, we consider uncertainties in M-TFM image parameters reconstruction in both numerical and experimental data to reproduce the on-site inspection. Additionally, we show how the trained neural network can learn its inner layers (i.e., the cU-Net layers) according to the physical parameters at stake so that it can be considered an open-box model enabling a qualitative interpretation of the generative process.

Keywords— ultrasonic array, total focusing method, weld inspection, multi-fidelity, deep learning, conditional UNet, generative models, meta-model

1 Introduction

Ultrasonic array imaging is among the most employed inspection methods for non-destructive testing and evaluation (NDT&E). This technology enables fast scanning of industrial components; thus, wider inspection areas can be covered in a reasonable amount of time. In addition, ultrasonic arrays are rather flexible since they allow different acquisition modalities (that could be done in parallel), such as acquisitions with focused or steered plane waves or the so-called full matrix capture (FMC). The latter consists of recording inter-element signals corresponding to all transmitter-receiver pairs in the array.

The most common FMC data post-processing algorithm is the Total Focusing Method (TFM) imaging, which provides optimal focusing and spatial resolution throughout the region of interest if compared to other delay-and-sum methods based on focused beams, such as in [1]. Therefore, ultrasound array is widely adopted in complex inspection scenarios, such as in weld inspection, where mechanical properties, flaw topology, and geometrical profile represent challenging issues to be taken into account for analysing the acquisitions. Furthermore, the Multi-modal Total Focusing Method (M-TFM) version [2–4] has been proven to be even more powerful in detecting and characterising different types of defects appearing at different positions in welds. Indeed, the possibility of considering multiple wave paths allows it to reach different locations within the weld where defects typically appear. M-TFM consists in exploiting different

images of a given defect from the same set of FMC data, each corresponding to a particular wave path before and after interaction with the defect. In the context of crack-like defects imaging with quasi-vertical orientations, the reconstruction modes can be classified into two categories: those exploiting the diffraction echoes from the defect (e.g., direct, indirect modes) and those which exploit the specular echoes from the quasi-vertical face of the crack (e.g., half-skip, indirect modes) with ultrasonic paths including at least one reflection with the specimen interfaces, as well as the mode conversions (see Fig. 1).

In practice, although the multi-modal approach can be useful to improve the inspection of welds, the quality of M-TFM images is strongly influenced by the actual knowledge of the phase velocities of longitudinal and transverse waves, the geometrical profile of the back-wall, as well as parameters of the experimental set-up (e.g., the position of the probe relative to the region of interest, water column height in the case of an immersion inspection). Uncertainties in such parameters can lead to errors in defect sizing and location, as well as imaging artefacts that make images difficult to interpret [5]. Furthermore, random noise sources coming from the acquisition and digital-to-analog conversion of signals (electronic noise) and from the wave scattering by heterogeneities in the material (structure noise) may impact the imaging. The lack of knowledge about the aforementioned influential parameters can be partially mitigated by using adaptive TFM schema to deal with irregular geometries [6] or by employing optimisation strategies for anisotropic materials with uncertain elastic proprieties [7] with payback in terms of reconstruction efficiency. Due to the challenges mentioned above, the analysis of M-TFM images in complex inspection problems is not straightforward and might require numerical solvers with a non-negligible computational burden.

Very recently, simulation-driven inversion strategies have been applied with success in NDT&E, mixing Machine Learning (ML), and Ultrasonic Testing (UT) [8, 9]. Those strategies based on time-domain signals or imaging data have also been studied in the context of structural health monitoring [10–16], as well as as in other domains such as eddy current testing [17–19]. The possibility to handle both numerical and experimental data and combine them in a tailored learning algorithm increases the efficiency of the deep learning (DL) algorithms, conceived to detect and size defects [12, 20–22]. In [23], the authors proposed a DL encoder-decoder architecture dealing with FMC data in order to suppress artefacts in reconstructed TFM images automatically. Once FMC experimental data are provided as inputs to the encoder-decoder, the DL architecture automatically performs denoising, reducing the presence of artefacts in TFM images.

DL has been adopted for data-augmentation strategies in [24]. It is worth noting that for the problems mentioned above, even the most accurate numerical solvers cannot fully reproduce some patterns that appear in experimental M-TFM images in a suitable computational time (i.e., a too-large combination of factors should be considered). This is often due to uncertainties or the lack of knowledge of input parameters.

The present study aims at developing a fully-controlled ML generative model targeting high-dimensional UT inspection problems embedding uncertainties coming from the lack of knowledge on inspection parameters and aleatory noise sources coming from unknown factors (e.g., electronic or structural noise, etc.). To this end, we propose a supervised learning schema based on physics-driven data issued from simulation tools aiming at replacing forward solvers in the massive and controlled generation of data used in advanced and time-consuming studies. In our analysis, we showed how a tailored cU-Net architecture enables high-quality multi-fidelity (MF) M-TFM reconstructions based on both numerical and experimental data. Moreover, we provide an analysis of the architecture's inner working structure by showing how the regression procedure is performed.

This paper is structured as follows: Section 2 exposes the principle of M-TFM used to produce the high and low fidelity data-set. Section 3 describes the architecture and the basis of the inner layers. In the same section, the loss function and the performance evaluations are summarised. The section 4 shows the implementation of this framework on the M-TFM data. A description of data production is given, along with the hyper-parameters chosen for the architecture in this section. Additionally, an exploration of the inner activation of the trained DNN is commented. The conclusions are summarised in Section 5. A discussion on the perspectives and possible applications are exposed in 6. We have made the code publicly available, along with a pre-trained network[1]. An accompanying video can be found

[1]https://github.com/geragranados/M_TFM_cUNet

under the same link.

2 Data

This section first recalls the principle of M-TFM imaging and its application to the inspection of welds where crack-like defects may appear and propagate along the chamfer. In addition, we describe the reconstruction modes that will be exploited to evaluate the constraint generative model procedure, and we provide some insight on the impact of reconstruction uncertainties on the M-TFM images.

The multi-modal imaging use-case considered in this paper is shown in Fig. 1b. It is a common NDT configuration for butt welds where an ultrasonic array is attached to a Rexolite wedge to perform an oblique inspection with either incident longitudinal (L) or transverse (T) waves depending on the wedge angle. The complex geometry of the steel mock-up is representative of a butt V-shaped weld with tilted interfaces on both sides of the weld root. A potential critical defect that can occur in welded parts is a crack that may propagate near or along the weld chamfer, and it is represented by a notch in the mock-up. A Region Of Interest (ROI) for the Eq. 1 is defined in the surrounding of the defect. A major challenge in NDE is the detection and characterisation of such defects as early as possible before their growth threatens the structural integrity of the component. It is precisely for this purpose that TFM has been extended to multi-modal imaging.

TFM is a "delay-and-sum" algorithm applied to the set of N^2 inter-element signals obtained from the FMC acquisition schema with an array of N elements. If $s_{nm}(t)$ is the signal received by element m when element n is used as a transmitter, the image amplitude at a given point located by $\mathbf{r}$ consists in a coherent sum of N^2 analytic signals $\hat{s}_{nm}(t) = s_{nm}(t) + iH\{s_{nm}(t)\}$ at appropriate propagation times $t = \tau^p_{nm}(\mathbf{r})$ where H denotes the Hilbert transform. $\tau^p_{nm}(\mathbf{r})$ is the theoretical time of flight between transmitter n and receiver m through the image point at $\mathbf{r}$ for the p^{th} reconstruction mode, i.e., for one of the many potential ultrasound paths that can be exploited for form relevant images of a given crack-like defect. With these notations, the image amplitude $I^p(\mathbf{r})$ can be calculated as

$$I^p(\mathbf{r}) = \left| \sum_{n=1}^{N} \sum_{m=1}^{N} \hat{s}_{nm}\left(\tau^p_{nm}(\mathbf{r})\right) \right|. \tag{1}$$

Multi-modal TFM is the formation of several images of the same defect from a single FMC data-set with appropriate modes, and these images can be combined to obtain a more realistic representation of the defect [4, 25–27]. These views aim to utilise ray paths that maximise the viewing angle of any particular defect, making visible defect responses across these views more likely.

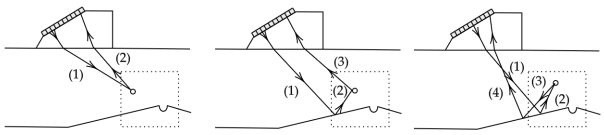
When no reflection on the back-wall occurs along the path between the elements and the running point P (Fig. 1a), the mode is called direct mode, in contrast with the half-skip and indirect modes, characterised by one or multiple reflections from the upper and lower surfaces. The combination of longitudinal (L) and transverse (T) waves leads to four direct reconstruction modes. L-L (or T-T) designates the mode for which the return trip is with longitudinal wave (or transverse wave), while L-T and T-L designate the modes with conversion when the incident wave-field interacts with the defect [2]. Additionally, half-skip modes denoted $P_1P_2 - P_3$ and indirect modes denoted $P_1P_2 - P_3P_4$ can be considered, where P_i = L o T (i.e., see 1a).

The subset of M-TFM images under the selected modes (direct, half-skip, and indirect) used in this work are shown in Fig. 2. The inspection is done in a complex specimen such as the one shown in Fig. 1b, where an artificial slot is considered a root-like crack.

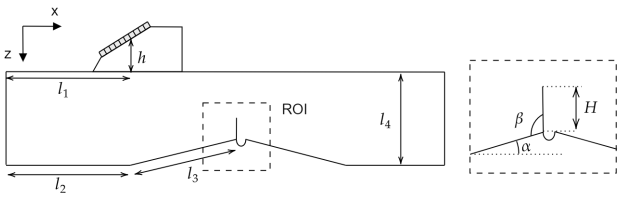
In this work, the numerical simulation of the UT array inspection represents the low-fidelity data. Meanwhile, the experimental mock-up acquisitions represent high-fidelity data. In both cases, the M-TFM imaging technique is applied to the temporal data to produce the image data-set. As a result, a MF data-set of M-TFM images is then defined as $\mathcal{D} = \{(\boldsymbol{x}_n, \boldsymbol{p}_n, \boldsymbol{y}_n) : n \in \{1, ..., M\}\}$, where M is the number of samples (Eq. 2). An instance $(\boldsymbol{x}_i, \boldsymbol{p}_i, \boldsymbol{y}_i)$ represents a low fidelity image $\boldsymbol{x}_i$ with labels $\boldsymbol{p}_i$ (simulation parameters) and its high fidelity couple $\boldsymbol{y}_i$, respectively. Couples of images are created by respecting the same parameters for both fidelities. The data-set described is then separated into training, validation, and test set. $\mathcal{D}$ contains:

$$\boldsymbol{x} \in \mathcal{X} \subset \mathbb{R}^{W \times H \times C}, \quad \boldsymbol{p} \in \mathcal{P} \subset \mathbb{R}^{n_p};$$

$$\boldsymbol{y} \in \mathcal{Y} \subset \mathbb{R}^{W \times H \times C}, \tag{2}$$

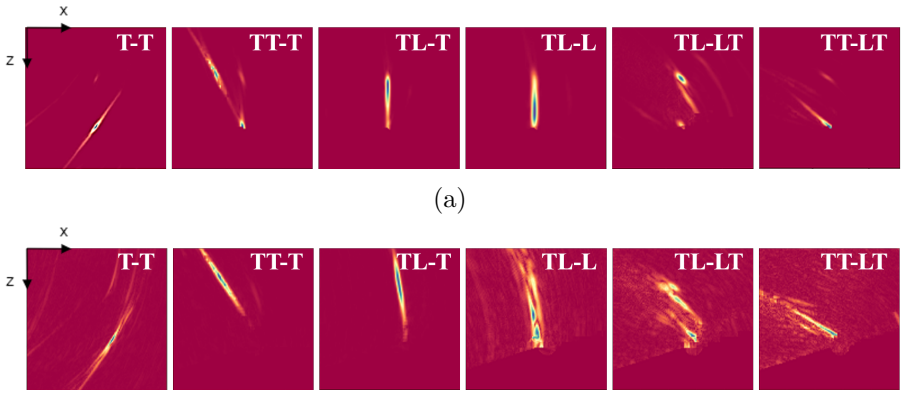


(a)



(b)

Figure 1: (a) Ray paths associated with three families of reconstruction modes: direct modes including two sub-paths; half-skip modes with three sub-paths; and indirect modes with four sub-paths. Each ultrasonic sub-path corresponds to the propagation of L or T waves. (b) Contact inspection configuration with a Rexolite wedge for imaging a back-wall breaking notch machined in a steel mock-up representative of a butt weld.



(a)

(b)

Figure 2: Examples of experimental M-TFM images with set-up shown in Fig. 1a. M-TFM images of the back-wall breaking notch are given for direct-, half-skip, and indirect mode reconstruction with (a) actual specimen given a celerity and flaw geometry and (b) same specimen with uncertainties in celerity T (cT) and α. Upper row: $cT = 3230$ m/s and $\alpha = 14°$. Lower row: $cT = 3380$ m/s and $\alpha = 18°$. The images are show in a normalised linear scale.

where W, H, C are the weight, height and number of channels for the images in both fidelities, and n_p size of the vector of parameter associated to each couple of images $\boldsymbol{x}_i$ and $\boldsymbol{y}_i$.

3 Method

In this paper, we propose to tackle the problem of generation of high-quality M-TFM images by employing a meta-model that accounts for simulations and experimental measurements together. To this end, a

supervised learning strategy relying on the use of a Deep Neural Network (DNN) architecture has been developed. More precisely, an end-to-end Conditional U-Net (cU-Net) such as those presented in [28, 29] has been conceived and developed. The DNN is designed to account for both numerical and experimental M-TFM images. The cU-Net architecture is trained with both simulations as low-fidelity data source and experimental measurements as high-fidelity data source along with physical parameters associated with the numerical set-up considered (e.g., flaw(s) size and orientation, specimen geometry, phase velocities of elastic waves, M-TFM reconstruction parameters, etc.) that plays the role of conditioning variables. Furthermore, to properly address the complexity of the targeted problem, the proposed cU-Net exploits advanced deep learning constitutive blocks such as the Fidelity Linear Modulation (FiLM), proposed by [30, 31], and a modified version of the spatial transformer block [32], named in this work as Parametric Spatial Transformer (pST). Thus, our cU-Net model can be considered as a MF model [33] aiming at enhancing the numerical simulation capabilities in terms of both accuracy and efficiency thanks to an almost real-time M-TFM images generation.

It is worth to be pointed out that, in contrast to other DL and ML close-box models based on a purely data-driven approach, the MF model developed in this work can be considered to be closer to an open-box since the physics-based knowledge is injected into the learning procedure via simulations. The objective of this strategy is to teach the surrogate model to encode all the useful information from the simulated M-TFM images while generating images close to the experience. At the same time, the encoder is intended to learn how its latent features are influenced by the simulation parameters to be able to rapidly reproduce several M-TFM images spanning the space of parameters.

3.1 Model architecture

The cU-Net is a modified architecture from the original U-Net [34]. The classical U-Net is essentially an encoder-decoder structure with skip-connections across the encoding and decoding trunks. Our cU-Net (Fig. 3) has as input an simulated image and its labels. The architecture is intended to be used as a parametric surrogate model and a realistic data generator, since the input is expected to be a experimental image. The objective is to learn the link between the simulation parameters (labels) $\boldsymbol{p}$ and the latent features $\boldsymbol{z} \in \mathcal{Z} \subset \mathbb{R}^{n_z}$ associated to the input image $\boldsymbol{x}$, although learning a disentangled latent manifold still represents a major scientific challenge [35]. In this sense, the trained cU-Net will output new realistic samples $\hat{\boldsymbol{y}}$ from the one simulated image $\boldsymbol{x}$, based on an excursion on $\boldsymbol{p}$ (considered as input labels).

This architecture was conceived to learn from a multi-class data-set, such as a database of M-TFM images for different mode reconstruction or different types of flaw geometry. The cU-Net is fed with batches of data $(\boldsymbol{x}, \boldsymbol{p})$ of size B. The model is expected to map $(\boldsymbol{x}_i, \boldsymbol{p}_i)$ to $\boldsymbol{y}_i$. The first part of the network $F : \mathcal{X} \rightarrow \mathcal{Z}$ (yellow blocks in Fig. 3) encodes the images $\boldsymbol{x}$ into a conditioned latent space representation $\boldsymbol{z}$ (features map of green block in Fig. 3). This encoding is conditioned by $\boldsymbol{p}$.

The decoder or generator G (blue blocks in Fig. 3) takes the space $\mathcal{Z}$ and the skip-connections to reconstruct the images $\boldsymbol{y} \in \mathcal{Y}$. Additional convolutions layers (the so-called *synthesis* layers, depicted in violet in Fig. 3) are stacked on the top of the generator G to improve the reconstruction. In this study, both F and G are featured by a stack of convolutional layers, with Parametric Rectified Linear Unit (PReLU) activation functions [37], except for the synthesis block layer. PReLU set the LeakyReLU activation coefficient as a learnable parameter for the DNN.

In order to inform the latent manifold with the M-TFM parameters $\boldsymbol{p}$, a FiLM+pST block is inserted within each convolutional layer in the encoder F. The objective of these operators is to structure the latent representation at the end of the encoder by learning the relation of parameters (labels) and features in different levels of resolution. The encoder extracts a sequence of features at each layer from a given input $\boldsymbol{x_i}$, we denote the output of a layer k as $\boldsymbol{x}_k \in \mathbb{R}^{r_k \times r_k \times C_k}$. r_k is the resolution at k after the down-sampling from $k-1$ and C_k is the number of filters (or channels) for k. At the FiLM and pST layers, the $\boldsymbol{x}_{k+1}$ have the same filter resolution and number of channels as the input $\boldsymbol{x}_k$.

A FiLM layer learns new per-channel statistics by applying the scales $\boldsymbol{\gamma}_k$ and the bias $\boldsymbol{\beta}_k$ to a normalised representation of $\boldsymbol{x}_k$ (Fig. 4b). The values for $\boldsymbol{\gamma}_k$ and $\boldsymbol{\beta}_k$ are learnables functions of $\boldsymbol{p}$. FiLM modifies the relative importance of features for the subsequent convolution operation $k+1$. The inference from the labels $\boldsymbol{p}$ is performed by a Multi Layer Perceptron (MLP). Eq. (3) represents the operation

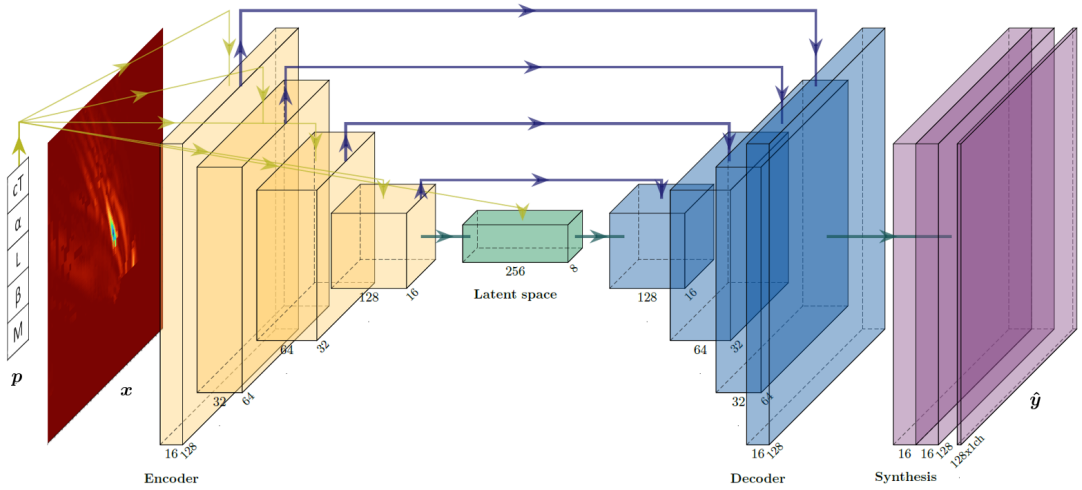


Figure 3: Schematic representation of cU-Net architecture, created with [36]. The input is a M-TFM image and its labels (simulation parameters). The DNN's encoder is represented in yellow, the latent space representation in green, and the decoder in blue. The violet layers represent the synthesis block. U-net skip-connections are shown by blue lines, and labels forward propagation on the encoder are represented by yellow arrows. An example of the number of filters and resolution at each block is given.

done per-channel (C_k) in the encoder, where $\boldsymbol{p}$ is the image label, $\boldsymbol{\gamma}_k(\boldsymbol{p})$ and $\boldsymbol{\beta}_k(\boldsymbol{p})$ are two output of MLP. FiLM$(\boldsymbol{x}_k, \boldsymbol{p})$ is the output of this layer, so the modified input features. In this layer, the mean and variance are computed per batch to normalise $\boldsymbol{x}_k$. ϵ is added for numerical stability and precision.

$$\text{FiLM}(\boldsymbol{x}_k, \boldsymbol{p}) = \boldsymbol{\gamma}_k(\boldsymbol{p}) \cdot \frac{\boldsymbol{x}_k - \mathbb{E}[\boldsymbol{x}_k]}{\sqrt{\sigma^2[\boldsymbol{x}_k] + \epsilon}} + \boldsymbol{\beta}_k(\boldsymbol{p}), \tag{3}$$

where $\boldsymbol{\beta}_k \in \mathbb{R}^{C_k}$ and $\boldsymbol{\gamma}_k \in \mathbb{R}^{C_k}$. $\mathbb{E}[\cdot]$ and $\sigma^2[\cdot]$ represent the empirical average and variance respectively.

Most of CNN applies a normalisation before each activation. The FiLM module replaces the normalisation layer (e.g., IN layer) in the encoding stream (F), giving the DNN a stable convergence and the degree of freedom to use the input $\boldsymbol{p}$ to improve the reconstruction and to allow the regression on the label space, as is shown in section 4.

Upon each FiLM layer in the encoder F, the proposed architecture stacked a pST layer. The original ST layer proposed by [32] was herein modified to make use of the parameter $\boldsymbol{p}$. An pST layer applies a spatial transformation capable of rotating, translating, and scaling a 2D input feature map $\boldsymbol{x}_k$ (Eq. (4) and Eq. (5)), via the per-instance matrix $T_{\boldsymbol{\phi}}$ that reads:

$$\text{pST}(\boldsymbol{x}_k, \boldsymbol{p}) = T_{\boldsymbol{\phi}}(\boldsymbol{p})\boldsymbol{x}_k, \tag{4}$$

where $\boldsymbol{\phi} \in \mathbb{R}^6$ and $T_{\boldsymbol{\phi}} \in \mathbb{R}^{3\times3}$, that reads:

$$\boldsymbol{\phi} = [\phi_1(\boldsymbol{p}),\ \phi_2(\boldsymbol{p}),\ \phi_3(\boldsymbol{p}),\ \phi_4(\boldsymbol{p}),\ \phi_5(\boldsymbol{p}),\ \phi_6(\boldsymbol{p})]$$

$$T_{\boldsymbol{\phi}}(\boldsymbol{p}) = \begin{bmatrix} \phi_1(\boldsymbol{p}) & \phi_2(\boldsymbol{p}) & \phi_3(\boldsymbol{p}) \\ \phi_4(\boldsymbol{p}) & \phi_5(\boldsymbol{p}) & \phi_6(\boldsymbol{p}) \\ 1 & 1 & 1 \end{bmatrix}. \tag{5}$$

Analogously, as in the FiLM layer, pST layer makes use of an MLP (known as localisation net) that infer $\boldsymbol{\phi} = \boldsymbol{f}_\theta(\boldsymbol{p})$ from $\boldsymbol{p}$ (Fig. 4a). The pST layer takes the input vector of parameters $\boldsymbol{p}$ to infer the spatial transformation for all channels. This transformation is applied to the output of the previous layer.

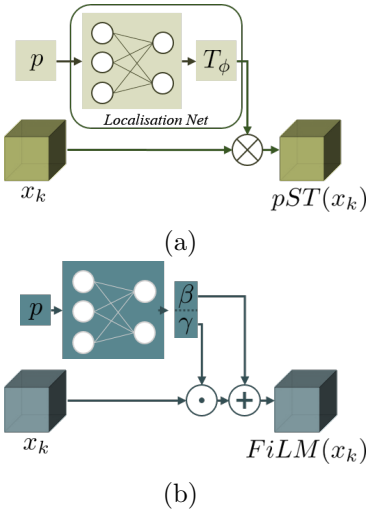


Figure 4: Schema of the parametric (a)FiLM and (b) pST layers. FiLM inner normalisation is not represented here for clarity.

On the other hand, the generator G makes use of Instance Normalisation (IN) [38] between each convolutional layer. IN showed impressive performance for feed forward stylising [39]. The DNN implements the pixel-wise L2-norm and the Focal Frequency Loss (FFL) [40] (Eq. (6)) as image reconstruction loss. The reconstruction loss $\mathcal{L}_{rec}$ reads:

$$\mathcal{L}_{rec}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{p}, \boldsymbol{\theta}) = ||\boldsymbol{y} - u_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{p})||_2 + FFL(\boldsymbol{y} - u_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{p})), \tag{6}$$

where $u_{\boldsymbol{\theta}}$ is the cU-Net and $\boldsymbol{\theta}$ its learn-able parameters.

The addition of the described modules is justified by the feature evolution observed on the images in $\mathcal{X}$ when exploring $\mathcal{P}$. For instance, a couple $(\boldsymbol{x}_i, \boldsymbol{p}_i)$ and a couple $(\boldsymbol{x}_j, \boldsymbol{p}_j)$ in the same reconstruction mode (e.g., T-T) are not far from each other in $\mathcal{X}$ if $\boldsymbol{p}_i$ and $\boldsymbol{p}_j$ are closed enough. Scaling, rotations, and translations in some features, together with changes in the echo shape, can be observed in $\boldsymbol{x}_j$ with respect to $\boldsymbol{x}_i$.

The transformations needed to obtain $\boldsymbol{x}_j$ from $\boldsymbol{x}_i$ are learnt by the module FiLM+pST in F. Therefore, the DNN is expected to set $F(\boldsymbol{x}_i)$ and $F(\boldsymbol{x}_j)$ near in the $\mathcal{Z}$ space. As a result, $u_{\boldsymbol{\theta}}$ serves as a parametric surrogate model. The DNN is expected to map the $\mathcal{P}$ space into the image feature space $\mathcal{X}$ and $\mathcal{Y}$ (injective mapping). For instance, if $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are two samples from the training set and for the test set, respectively; the couple $(\boldsymbol{x}_i, \boldsymbol{p}_j)$ must generate the instance $\boldsymbol{y}_j$ (with $\boldsymbol{p}_i \neq \boldsymbol{p}_j$), without need to know $\boldsymbol{x}_j$ during the training. The model does not implement any additional loss for this task besides $\mathcal{L}_{rec}$, so the $\mathcal{P}$ to $\mathcal{Y}$ conditional mapping is learnt thanks to two intrinsic characteristics of the DNN. The first characteristic is the FiLM+pST module introduced in the encoder F, which learns to condition the features extracted from the input image with different labels. The second one is related to the supervised framework adopted. That is, $\boldsymbol{x}_i$ and $\boldsymbol{y}_i$ have the same label $\boldsymbol{p}_i$ associated during the training since the data-set is presented by couples of instances from both fidelity levels. As a consequence, the output image $\hat{\boldsymbol{y}}_j$ can be directly labelled by input parameters $\boldsymbol{p}_j$. The results of this application are shown later in section 4.

On the other hand, $u_{\boldsymbol{\theta}}$ maps the space $\mathcal{X}$ into $\mathcal{Y}$ by learning the differences between the images from $\mathcal{X}$ and from $\mathcal{Y}$. Those differences can be appreciated in the Signal to Noise Ratio (SNR) variation, the flaw echo location and shape. The decoder G acts here as a realistic data generator.

Some metrics are implemented to measure the echo localisation error and the pixel intensity error for the couples $\boldsymbol{y}$ versus $\hat{\boldsymbol{y}}$. The metrics express the reconstruction quality for the DNN either for the task of realistic generation data or the parametric model application. The maximum value of the L1-norm for an instance expressed by Eq. (7) is used to quantify the error at the peak value, normally located in the echo region of the M-TFM image. Another metric is the Mean Absolute Error (MAE) which quantifies in average the reconstruction error for $\hat{\boldsymbol{y}}$. The position error of the maximum obtained by Eq. (7) is also a metric for evaluating the performance on the test set.

Table 1: Parameter space definition for M-TFM simulation and experimental data-set. Vertical: $\beta - \alpha = 90°$; Tilted: $\beta - \alpha = 74°$

Parameter space (labels)	**Range of values**	**n-points in $\mathcal{P}$**
Flaw tilt (β)	Vertical ; Tilted	2
Flaw size (L)	3 mm ; 10 mm	2
T wave celerity (c_T)	[3080; 3380]	9
Geometry configuration angle (α)	[10;18]	9
Modes used in reconstruction (M)	T-T;TT-T;TT-L;TL-T TL-L;TT-TT;TL-LT; TT-LT;ALL	9
	Total of instances: (couples)	2916 (324 per mode)

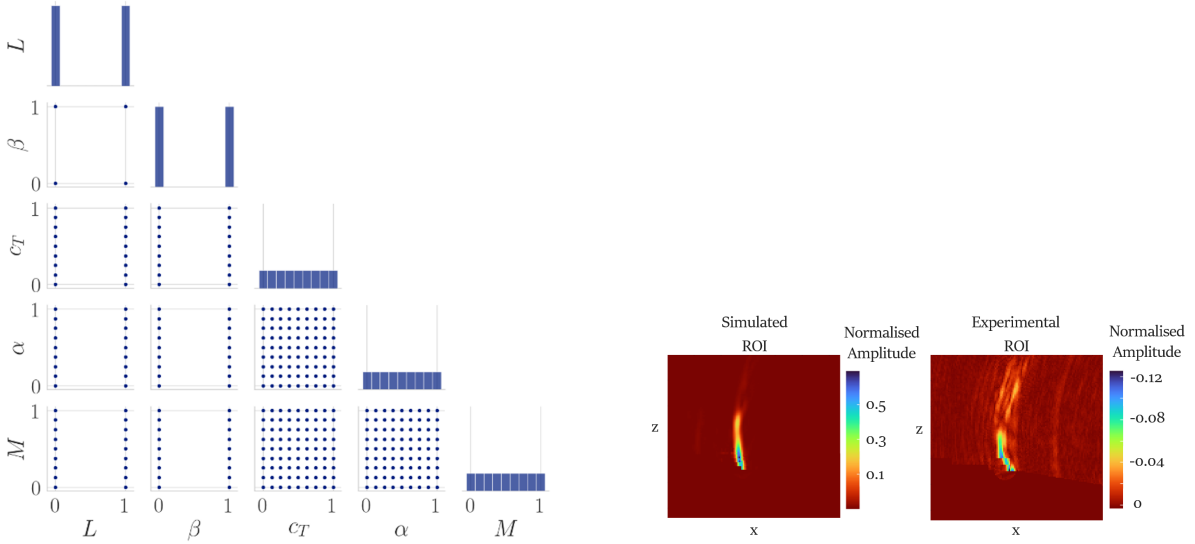
$$\mathcal{E}_{rec}(x_j, p_j, y_j) = \max ||y_j - u_{\boldsymbol{\theta}}(x_j, p_j)||_1 \tag{7}$$

4 Results

The present section illustrates the outcome of the trained architecture on the test set, in other words, images never seen by the DNN. Firstly, a realistic image generation is evaluated for the test set through the adapted metrics. Secondly, a conditional generation through regression in the input $\boldsymbol{p}$ is evaluated and compared to the realistic generation error. Lately, an exploration by a 2D projection of the output features of the latent space block is used to show the data generation potential. The shown structure of those features tends to explain how the DNN is capable of acting as a parametric surrogate model to generate new M-TFM instances. With the same objective of explaining how the DNN works, the exploration of the features extracted by the FiLM layer and the pST layer are shown to interpret their role in the architecture.

The presented architecture was tested in a M-TFM image data-set with two fidelities: simulation and experimental acquisitions. The data was obtained by a parametric simulation by sampling the $\mathcal{P}$ space as shown in Table 1. The mock-up in Fig. 1b follows the configuration given by [26], with a geometry described by $l_1 = 50$ mm; $l_2 = 60.5$ mm; $l_3 = 50$ mm; and $l_4 = 42$ mm, and a ROI of a square of 30 mm of length. Consequently, the data-set $\mathcal{D}$ of $M \sim 6$ k single-channel images is produced. Half of the images belong to simulated instances, while the second half is the experimental couples. The data-set is separated in the proportion of 85%, 12%, 3% for the training ($\mathcal{T}$), validation ($\mathcal{V}$) and test set ($\mathcal{S}$), respectively. A random selection of the group $(\boldsymbol{x}_i, \boldsymbol{p}_i, \boldsymbol{y}_i)$ is done to generate disjoint sets so $\mathcal{D} = \mathcal{T} \cup \mathcal{V} \cup \mathcal{T}$. The image size is (W, H, C) = (128 pixels, 128 pixels, 1 channels). The input label size is $d = 5$ containing the celerity of the transversal ultrasonic wave (c_T), the back-wall angle (α), the reconstruction mode (M), and the flaw geometry expressed by the height (L) and the tilt (β). Those labels are re-scaled to be expressed in the range of $[0, 1]$ (Fig. 5). For those labels that had semantic representation (e.g., reconstruction modes), a simple dictionary is created to map the names to numerical values in the same range as other numerical labels.

The number of channels per block is augmented by a power of 2 in the encoder F, starting with 16 channels after the input to ending up with 128 channels before the latent space block. The resolution of the learnt filters at the end of every block of F is decreased by the down-sampling operation. The resulting resolutions by block at the encoder are 128, 64, 32, and 16. Each resolution block contains two FiLM+pST layers; the first is sandwiched between a convolutional layer and its PReLU activation. The activation is followed by the second FiLM+pST layer that is set between the second convolution operation of the block and its PReLU activation. This layer sequence ends up with a down-sampling operation before passing to the next resolution block.



(a) (b)

Figure 5: The M-TFM data-set contains a set of images labelled by the simulation parameters and classes. (a) Sampling of parameters $\boldsymbol{p} \in \mathcal{P}$ (simulation parameters for each coupled of M-TFM images). The blocks represent the distribution of the sampling for each parameter and the dots gives a quick view of, for instance, cT and alpha parameter intervals of sampling and number of sampled values. Beta and L show, for example, that there are four defects in the data-set, since the points are four. (b) An example of M-TFM images, right: simulated ($\boldsymbol{x} \in \mathcal{X}$); left: experimental acquisition ($\boldsymbol{y} \in \mathcal{Y}$). The two images correspond to the same parameter's vector $\boldsymbol{p}$.

Regarding the latent space block, the structure is similar to an encoder block since it accounts for the FiLM+pST layer. The difference is that there is no down-sampling. As a consequence, the block is built as follows: FiLM+pST + PReLU + convolution layer + FiLM+pST + PReLU. The resolution at this level is 8x8 with 128 channels for all the layers. The objective of this arrangement is conditioning the encoding of the DNN throughout all layers until the first up-sampling in the decoder so the labels build a structured latent space for the generator's input.

For the decoder G, we inverted the order in terms of resolution and number of channels, keeping the same number of layers as the encoder. Instead of the down-sampling, the up-sampling operation at the end of each resolution block is used to increase the resolution. The feature normalisation is done by the IN layers before each activation layer on the decoder. Finally, the synthesis block has three convolutional layers with the same resolution as the expected output but with more channels in the first two convolution layers: 16 channels. This block also implements the IN, but it differs from the encoder's blocks since a hyperbolic tangent activation function is used on it.

4.1 Analysis of the training phase: the role of FiLM+pST

This section shows the convergence of the DNN driven by its architecture, particularly by the presence of the feature operators such as the FiLM and pST layers. The evolution of loss versus the epochs is shown in Fig. 6 for the validation and test. The batch size (B) is chosen equal to 128, and an Adam optimiser with a learning rate of 0.001 is chosen. The training time is $\sim$ 3h on an NVIDIA Quadro RTX 6000 GPU. The early stopping activated at around 900 epochs. The trained model is then applied to the test examples in the following sections.

During the training, we observed that adding the parameter of reconstruction modes (M) to the label vector helped the reconstruction quality and the convergence speed, even if a regression over these parameters does not have any physical sense (e.g., a hybrid T-T+TT-L mode). This parameter plays a role in the structure of the latent space and helps to the DNN convergence. In the opposite sense, the simple training of the DNN without any conditioning (no FiLM+pST blocks, no $\boldsymbol{p}$ in the input, so an U-Net architecture) gets the best reconstruction for $\hat{\boldsymbol{y}}$. Consequently, the conditioning worsens

the reconstruction capability, but it introduces the surrogate modelling potential into the architecture: generation of M-TFM conditioned by physical parameters.

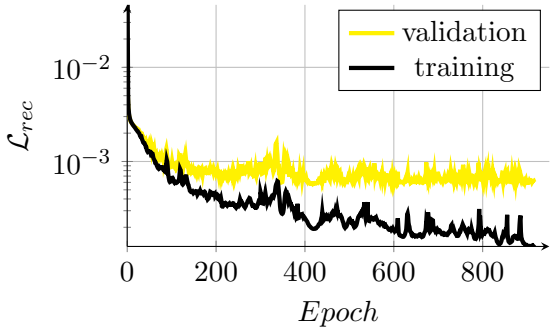
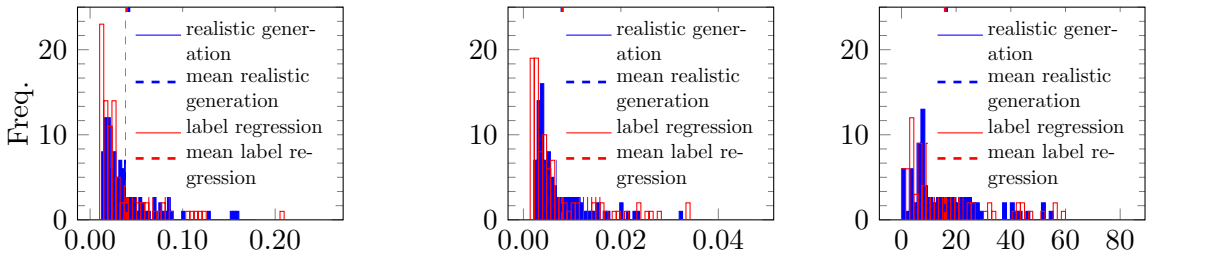


Figure 6: Loss evolution versus training epochs for cU-Net training on M-TFM data-set. The test loss is in black, and the validation loss is in yellow.

During the test phase, the $\mathcal{S}$ set is used to evaluate the DNN performance in order to show the capability of the cU-Net as a MF generator. The couples $(\boldsymbol{x}_j, \boldsymbol{p}_j) \in \mathcal{S}$ are forwarded from the input in the trained DNN $u_{\boldsymbol{\theta}_*}$ to generate $\hat{\boldsymbol{y}}_j \in \mathcal{S}$. Since $\boldsymbol{y}_j$ is our GT, we can compute the error histogram over the test set for echo reconstruction metrics introduced at section 3.1. Fig. 7 represents the echo amplitude error frequency for the 100 samples of the test set. The echo position error is also calculated by the Euclidean distance in the (x, z) plane of the ROI. In Fig. 8, a single instance reconstruction on the test set is shown in detail to illustrate the reconstruction error impact in a qualitative view.



(a) (b) (c)

Figure 7: Error frequency on the test set for realistic data-set generation in blue. Generation error frequency on reconstruction parameters regression are plotted in red (described in section 4.1). (a) shows the maximum of the pixel-wise difference between the output and the GT, (b) is the mean of the norm-L1 between the output and the GT, and (c) is the Euclidean distance in pixels between the maximum pixel of the output and the maximum pixel of the GT.

A second evaluation over the $\mathcal{S}$ set is done to test the generative capabilities of the DNN. This time, a $\boldsymbol{x}_i \in \mathcal{T}$ and a $\boldsymbol{p}_j \in \mathcal{S}$ are picked up. The couples $(\boldsymbol{x}_i, \boldsymbol{p}_j)$ are forwarded from the input in the trained DNN $u_{\boldsymbol{\theta}_*}$ to generate $\hat{\boldsymbol{y}}_j \in \mathcal{S}$. $\boldsymbol{x}_j$ is our Ground Truth (GT) labelled by $\boldsymbol{p}_j \neq \boldsymbol{p}_i$. Then, we can compute the error histogram over the test set on echo reconstruction metrics (Fig. 7). The results show the capability of the cU-Net as a surrogate model for new instances generation.

This test is a quantitative evaluation of data generated over a regression $\boldsymbol{p}$ input vector. Since only a regular sampling was done over the parametric $\mathcal{P}$ to create the entire data-set, the references of experimental M-TFM images for this evaluation are limited to the known $\boldsymbol{p}$ values. The selection of the instances of $\boldsymbol{x}_i$ (TFM simulation input) for this evaluation are chosen by the following criteria: the label $\boldsymbol{p}_i$ of $\boldsymbol{x}_i$ is the closest to $\boldsymbol{p}_j$, in the $\mathcal{P}$ space. The metric used in $\mathcal{P}$ is the Euclidean distance between $\boldsymbol{p}_i$ and $\boldsymbol{p}_j$. This metric promotes the generation by reconstruction parameters regression (c_T and α). Thus, a regression over the flaw geometry parameters (β and L) was only performed to evaluate the generation. Both tests reported similar histograms (red bars in Fig. 7). See the complementary video to illustrate conditional generation.

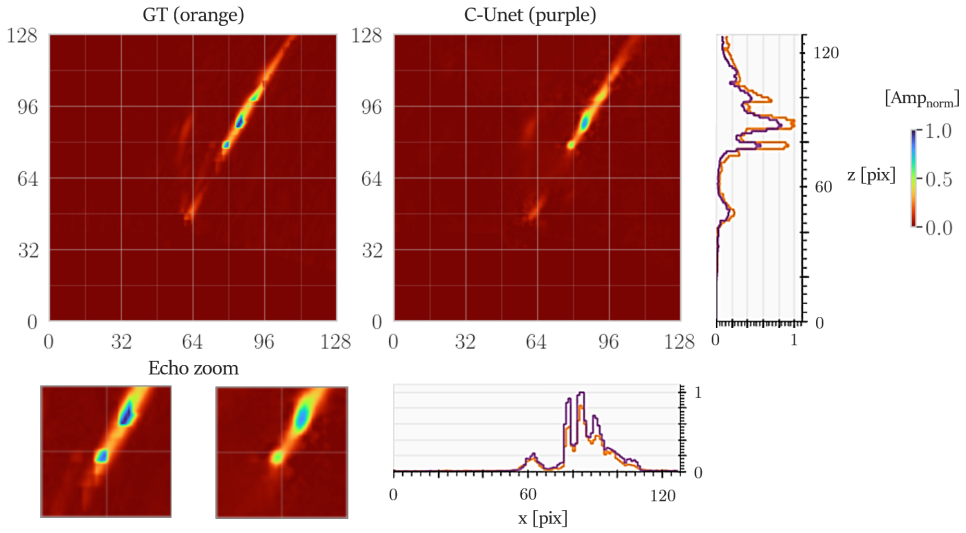


Figure 8: Experimental generation example. Qualitative comparison between the output $\hat{\boldsymbol{y}}$ and the GT, given a couple $(\boldsymbol{x}, \boldsymbol{p})$ at the input, where $\boldsymbol{p}$ is the correspondent set of parameters used to create $\boldsymbol{x}$ from the simulation. (a) is the maximum pixel for the line or column between the output and the GT. The maximum pixel amplitudes are plotted in purple for the output and in orange for the GT. Both images are overlapped for the ROI representation. In (b), the echo reconstruction shows the misfitting in terms of echo position and amplitude for the example given.

The described evaluation was designed to show the potential of the DNN for a surrogate model application. The model learns the link between the labels and the features required at the output, guided by the pST and FiLM layers. The followings sections are focused on the internal features exploration of the DNN for the test set with the intention of describing how the parametric regression can be done in all directions of $\boldsymbol{p}$ to re-sample the $\mathcal{P}$ beyond the initial data-set points.

4.2 Analysis of the feature maps

We observed how the conditional FiLM+pST layers act during the latent space generation when the scalar input p changes, an example of forward propagation of one image is shown in Fig. 9. The output for each column (purple box image) can be contrasted with its GT (below the column) to evaluate the coherence of the conditional generated image. This evaluation is done over a sample of the test-set.

We can observe that the conditional layers make the filter implant new features in the activation maps; at the same time, they extract the main features of the input image. The intuition here is that the encoder is capable of creating new conditional features on the activation maps from the input image; these new features are conditioned by the input label values. Consequently, the image synthesis is also conditioned by the input label values. We can infer that the variety of possible experience-like images (outputs) that the decoder can create is linked to the range of the parameters observed during the training (see Fig. 5).

Fig. 9 also shows the behaviour of the DNN without the pST layer. This is done to show the role of each block: FiLM acts in the style content (e.g., SNR), and the pST acts in the context (echo shape). The last column shows that the FiLM ignores the echo translation linked to the new cT input value. At the same time, we can observe how a roto-translation is present between the first and second column due to the pST module from the second convolutional operation.

4.3 Latent space exploration

A view into the output of the latent space block gives some clues about how the encoder works. When a M-TFM image is forwarded into the encoder, the features of the last layer of the latent space block represent a plausible high-dimensional representation of the input. This representation is obtained for each sample to generate the $\mathcal{Z}$ space. Since we are interested in the structure of this high dimensional

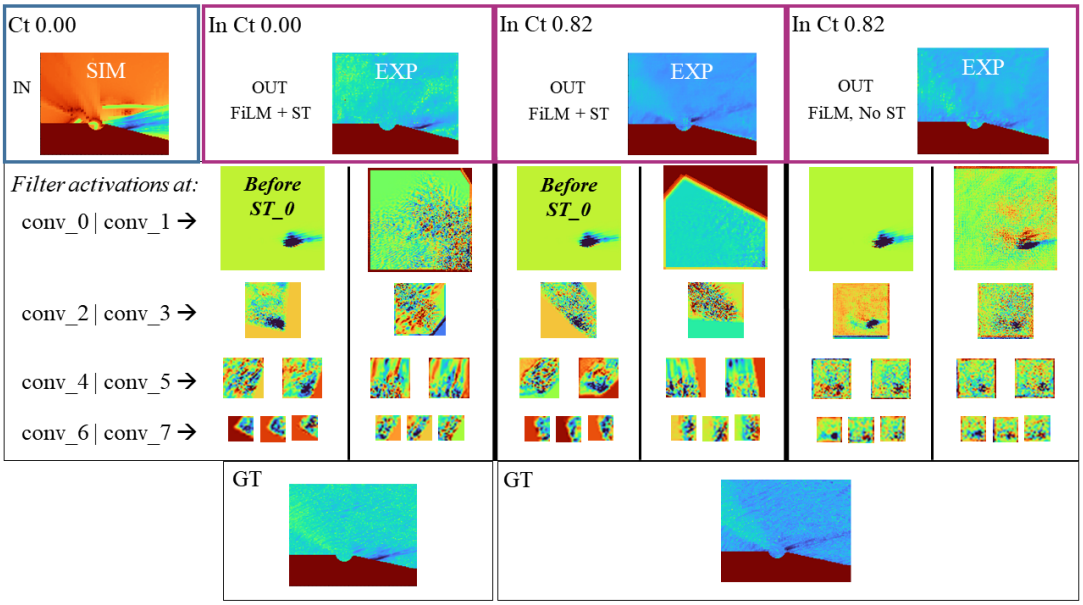


Figure 9: The first column shows the activation values for the filters at different resolutions on F when the realistic generation is evaluated. At the end of the column, the GT image is shown for a qualitative observation. To better appreciate the changes in style, the input and output images are reported in logarithmic scale. The activation maps are normalised for better visualisation. For the second column, a conditional generation by changing the cT input value is evaluated. In this case, the same input as the first column is kept. For the third column, the same parameter was varied (cT), but this time by ablation of the pST blocks.

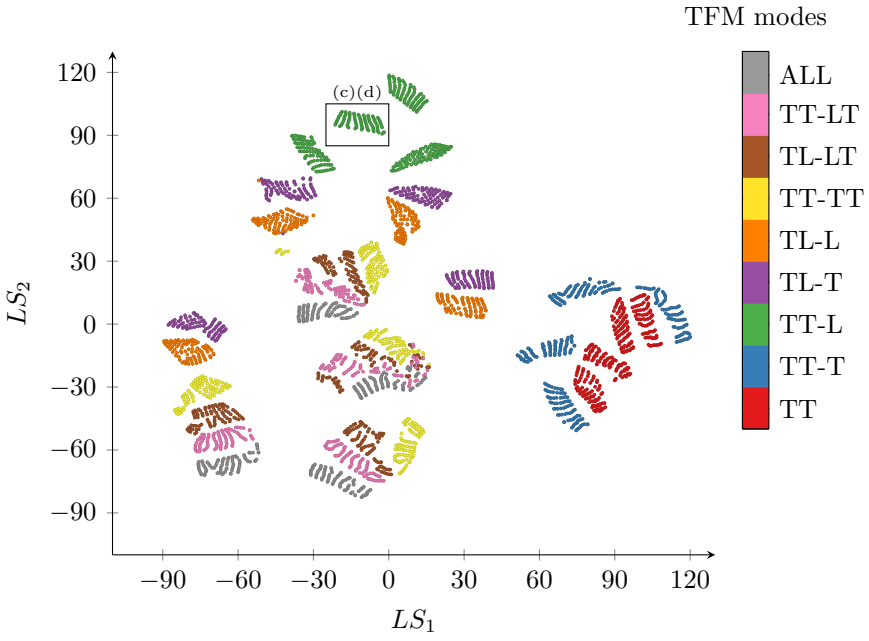
space (i.e., 8x8x256 dimensions), we choose a t-SNE manifold projection [41] to reduce this space to a latent space (LS) representation in 2D. A principal component analysis initialisation is used for the t-SNE.

The Fig. 10 represents a scatter plot of the $\mathcal{Z}$ space. A projection from $\mathcal{X}$ conditioned on $\mathcal{P}$ space to $\mathcal{Z}$ space is done by the encoder. Every $\boldsymbol{z}$ point is located on the 2D manifold by its coordinates (LS_1, LS_2). Consequently, each point can be related to a parameter value and input image. The representation in Fig.10 shows how the DNN builds $\mathcal{Z}$ in terms of clusters ordered accordingly to the different parameters.

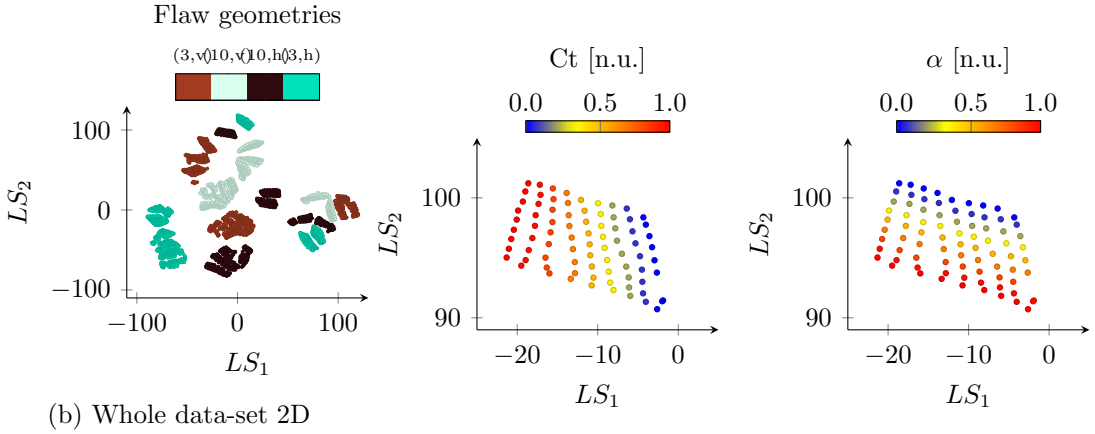
That is, the first hierarchy observed in the manifold is the reconstruction modes. This order in the hierarchy is expected since the images in each reconstruction mode are significantly different. If we observe the parameters of the flaw's geometry, four main clusters are founded in the manifold. When looking closer at one of these manifolds, an arrangement with respect to the two last parameters in $\mathcal{P}$ can be observed. Two mostly orthogonal directions are present for the celerity T and back-wall angle values.

The exposed structure of the $\mathcal{Z}$ space in this section, together with the exploration of the features, shed some light on how the DNN learns to generate data. The observations proved that the DNN creates a structured representation of the whole data-set. The $\mathcal{Z}$ sampling in the right direction with respect to the parameters of the simulation is possible through the conditional encoder. This is how the input vector can generate other samples present in the data-set, but also some samples not seen before. Those samples generated by the label regression are obtained by a consistent position in $\mathcal{Z}$. During the generation by regression, the encoder features are modified in the coherent direction, as is shown in the $\mathcal{Z}$ projection.

Given a fixed simulation image and a new label value never seen, the encoded "point" in the latent space is then passed to the generator. The generators learnt to decode the known points of $\mathcal{Z}$, as it was shown in the section 4.1. It is also expected for the generator to know "how" to generate an intermediate point with similar characteristics to the neighbours but modified to be consistent with the new label value.



(a) Whole data-set 2D projection, colored by modes



(b) Whole data-set 2D projection, coloured by flaw geometries

(c) Rectangle in (a) coloured by normalized values of cT

(d) Rectangle in (a) coloured by normalized values of α

Figure 10: c-Unet latent space projection into a 2D t-SNE manifold. The point represents the last layer activation of the green block in the c-UNet (Fig. 3), just before the decoding. (a) shows the clustering by modes. (b) shows the clusterig by flaw geometries. (c) and (d) is the a selected cluster (black rectangle in (a)) colored by two of the parameters: cT and α. t-SNE hyperparameters: early exageration=12, perplexity = 50. References for flaw geomatries coloring: (3,h) is the 3 mm horizontal flaw geometry, (10,v) is the 10mm vertical flaw geometry, and so on.

5 Conclusions

In this work, we present a conditional UNet (cU-Net) for fast and realistic generation of multi-modal total focusing method (M-TFM) images. Our DNN is trained on both numerical and experimental data. As a result, our generative framework can learn realism from experiments along with a controlled generation learnt from the physics encoded in the simulations. We show how the cU-Net model performs the feature extraction to generate new realistic data. This is done by exploring the inner activation layers for different inputs. We also demonstrate how the Spatial Transformer Layer and Fidelity Layer

Modulator provide the mean to perform a regression in the M-TFM images by a cU-Net architecture.

The present framework allows the inclusion of simulation parameters directly as input in the meta-model architecture. Once the model is trained, data generation can be done in quasi-real-time and guided by the input parameters regression. In doing so, the generated data is already labelled. However, the prior information on data fidelity (simulation and experience labelled couples) can be costly to produce in some cases, and our approach turns out to be not highly efficient when this characteristic is not available in the data-set. Nonetheless, the present architecture provides a way to fully inform a data generator DNN by the data labels.

Given this choice in the DNN architecture used for TFM image generation, we are not in presence of a physics-agnostic NN. Moreover, the DNN is constrained to learn the physics behind the mapping between the simulated parameters and the TFM images. This is done by the pST and FiLM operators, as showed in the Fig. 9 by the ablation of the pST during the generation of new images with different celerities. The reconstruction loss plays also a role for this task during the training, since the conditional layers are optimized to generate a consistent experience image given a set of parameters. To summarize, we embed the physics knowledge in the DNN architecture by means of the data simulated by the numerical solver, and the conditional generation driven by the parameters used in the simulation. Our approach is in contrast to more common data-driven approaches that are in the literature of ML as applied to image or signal processing communities.

An extensive analyses of the results by changing simulation parameters has been embedded in the Appendix A.

Scalar labels were used as input in this work, so the inner sub-neural networks for the FiLM and pST Layer are Multi Layer Perceptron (MLP)s. Those sub-neural networks are not limited to scalar inputs, but they can any shape. For instance, a CNN or RNN can be deployed to take into account other formats of labels (images, time series, etc.) that would guide later the generation.

6 Perspectives and applications

Fast and reliable surrogate models has been used in NDT&E to speed-up the computational time for very intensive statistical studies ranging from the stochastic inversion [42] to the sensitivity analysis [33] and model-assisted probability of detection. In these application fields in NDT&E, our contribution aims to enhance the quality of meta-model results making them more-close-to-reality. Therefore, by the use of the ML schema developed, the aforementioned studies will better account the impact of measure-like uncertainties in the studies outcomes.

This approach presents a first attempt at a physics-based and explainable meta-model aiming at providing a controllable data generation tool. The new data can be exploited as a training set for deeper architectures (e.g., generative adversarial network) that demands big data-sets, rarely available in the NDT&E field. Inversion problems may also benefit from an enlarged training set generated by this tool.

In our application case, we include the MF knowledge present in the data-set, but the framework can be adopted even if a single fidelity level is available in an auto-encoder architecture (no skip connections required). For instance, the same image may be used as input and output to train the cU-Net and get a meta-model from a simulated data-set. Likewise, an experimental campaign may be enlarged by this DNN when it is correctly labelled.

In the perspectives of this work, we expect to explore the power of our cU-Net backbone architecture for inverse problems, given that structure of the latent space seems promising for this task. Additionally, one notable mention is the possibility of using the cU-Net architecture to enhance the performance of simulation software widely used in modern NDT&E design problems. For instance, the low fidelity source of data may come from coarse meshed finite elements simulations or semi-analytical simulations, whereas the high fidelity from fine meshed finite elements simulations, among others.

Author Credits

- GEG: Conceptualization. Methodology. Software. Visualization and data presentation. Writing-Original draft preparation.
- RM: Conceptualization. Methodology. Data curation. Writing- Original draft preparation.
- FG: Conceptualization. Methodology. Writing- Reviewing and Editing.
- SR: Data collection. Writing- Original draft preparation.

- DC: Conceptualization.

Acknowledgements

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 800945-NUMERICS-H2020-MSCA-COFUND-2017

References

[1] Caroline Holmes, Bruce W. Drinkwater, and Paul D. Wilcox. Post-processing of the full matrix of ultrasonic transmit–receive array data for non-destructive evaluation. *NDT & E Int.*, 38(8):701–711, December 2005.

[2] Lucas Merabet, Sébastien Robert, and Claire Prada. The multi-mode plane wave imaging in the Fourier domain: Theory and applications to fast ultrasound imaging of cracks. *NDT & E Int.*, 110:102171, March 2020.

[3] Léonard Le Jeune, Sébastien Robert, Eduardo Lopez Villaverde, and Claire Prada. Plane Wave Imaging for ultrasonic non-destructive testing: Generalization to multimodal imaging. *Ultrasonics*, 64:128–138, January 2016.

[4] Rhodri L. T. Bevan, Nicolas Budyn, Jie Zhang, Anthony J. Croxford, So Kitazawa, and Paul D. Wilcox. Data fusion of multiview ultrasonic imaging for characterization of large defects. *IEEE Trans. Ultrason. Ferroelectr. Freq. Control*, 67(11):2387–2401, 2020.

[5] Stéphane Le Berre, Xavier Artusi, Roberto Miorelli, Ekaterina Iakovleva, and Pierre Calmon. Simulation and processing tools for the design and performance evaluation of FMC-TFM techniques. *AIP Conf. Proc.*, 2102(May), 2019.

[6] Leonard Le Jeune, Sébastien Robert, P Dumas, A Membre, and Claire Prada. Adaptive ultrasonic imaging with the total focusing method for inspection of complex components immersed in water. In *AIP Conf. Proc.*, volume 1650, pages 1037–1046, 2015.

[7] Corentin Ménard, Sébastien Robert, Roberto Miorelli, and Dominique Lesselier. Optimization algorithms for ultrasonic array imaging in homogeneous anisotropic steel components with unknown properties. *NDT & E Int.*, 116:102327, 2020.

[8] Sergio Cantero-Chinchilla, Paul D Wilcox, and Anthony J Croxford. Deep learning in automated ultrasonic nde–developments, axioms and opportunities. *NDT & E Int.*, page 102703, 2022.

[9] Hongbin Sun, Pradeep Ramuhalli, and Richard Jacob. Machine learning for ultrasonic nondestructive examination of welding defects: A systematic review. *Ultrasonics*, page 106854, 2022.

[10] Jiaxing Ye, Shunya Ito, and Nobuyuki Toyama. Computerized ultrasonic imaging inspection: From shallow to deep learning. *Sensors*, 18(11):3820, 2018.

[11] Nauman Munir, Hak-Joon Kim, Jinhyun Park, Sung-Jin Song, and Sung-Sik Kang. Convolutional neural network for ultrasonic weldment flaw classification in noisy conditions. *Ultrasonics*, 94:74–81, 2019.

[12] Richard J. Pyle, Rhodri L.T. Bevan, Robert R. Hughes, Rosen K. Rachev, Amine Ait Si Ali, and Paul D. Wilcox. Deep learning for ultrasonic crack characterization in nde. *IEEE Trans. Ultrason. Ferroelectr. Freq. Control*, 3010(c), 2020.

[13] Luca Rosafalco, Andrea Manzoni, Stefano Mariani, and Alberto Corigliano. Fully convolutional networks for structural health monitoring through multivariate time series classification. *Adv. Model. Simul. Eng. Sci.*, 7(1):1–31, 2020.

[14] Luka Posilović, Duje Medak, Marko Subašić, Marko Budimir, and Sven Lončarić. Generative adversarial network with object detector discriminator for enhanced defect detection on ultrasonic b-scans. *Neurocomputing*, 459:361–369, 2021.

[15] Long Bai, Florian Le Bourdais, Roberto Miorelli, Pierre Calmon, Alexander Velichko, and Bruce W Drinkwater. Ultrasonic defect characterisation using the scattering matrix: A performance comparison study of bayesian inversion and machine learning schemas. *IEEE Trans. Ultrason. Ferroelectr. Freq. Control*, 2021.

[16] Tuomas Koskinen, Iikka Virkkunen, Oskar Siljama, and Oskari Jessen-Juhler. The effect of different flaw data to machine learning powered ultrasonic inspection. *J. Nondestruct. Eval.*, 40(1):1–13, 2021.

[17] Shamim Ahmed, Christophe Reboud, Pierre-Emile Lhuillier, Pierre Calmon, and Roberto Miorelli. An adaptive sampling strategy for quasi real time crack characterization on eddy current testing signals. *NDT & E Int.*, 103:154–165, 2019.

[18] Peipei Zhu, Yuhua Cheng, Portia Banerjee, Antonello Tamburrino, and Yiming Deng. A novel machine learning model for eddy current testing with uncertainty. *NDT & E Int.*, 101:104–112, 2019.

[19] Robert R Hughes and Bruce W Drinkwater. Exploring high-frequency eddy-current testing for sub-aperture defect characterisation using parametric-manifold mapping. *NDT & E Int.*, 124:102534, 2021.

[20] Roberto Miorelli, Clement Fisher, Andrii Kulakovskyi, Bastien Chapuis, Olivier Mesnil, and Oscar D'Almeida. Defect sizing in guided wave imaging structural health monitoring using convolutional neural networks. *NDT & E Int.*, page 102480, 2021.

[21] Thulsiram Gantala and Krishnan Balasubramaniam. Automated Defect Recognition for Welds Using Simulation Assisted TFM Imaging with Artificial Intelligence. *J. Nondestruct. Eval.*, 40:28, 2021.

[22] Richard J. Pyle, Rhodri L.T. Bevan, Robert R. Hughes, Amine Ait Si Ali, and Paul D. Wilcox. Domain Adapted Deep-Learning for Improved Ultrasonic Crack Characterization Using Limited Experimental Data. *IEEE Trans. Ultrason. Ferroelectr. Freq. Control*, 69(4):1485–1496, 2022.

[23] Sergio Cantero-Chinchilla, Paul D Wilcox, and Anthony J Croxford. A deep learning based methodology for artefact identification and suppression with application to ultrasonic images. *NDT & E Int.*, 126:102575, 2022.

[24] Iikka Virkkunen, Tuomas Koskinen, Oskari Jessen-Juhler, and Jari Rinta-Aho. Augmented ultrasonic data for machine learning. *J. Nondestruct. Eval.*, 40(1):1–11, 2021.

[25] Evgeny G. Bazulin. On the possibility of using the maximum entropy method in ultrasonic nondestructive testing for scatterer visualization from a set of echo signals. *Acoust. Phys.*, 59(2):210–227, March 2013.

[26] Kombossé Sy, Philippe Brédif, Ekaterina Iakovleva, Olivier Roy, and Dominique Lesselier. Development of the specular echoes estimator to predict relevant modes for Total Focusing Method imaging. *NDT & E Int.*, 99:134–140, 09 2018.

[27] Kombossé Sy, Philippe Brédif, Ekaterina Iakovleva, Olivier Roy, and Dominique Lesselier. Development of methods for the analysis of multi-mode tfm images. In *J. Phys.: Conf. Series*, volume 1017, page 012005. IOP Publishing, 2018.

[28] Gabriel Meseguer-Brocal and Geoffroy Peeters. Conditioned-U-Net: Introducing a control mechanism in the U-net for multiple source separations. *Proc. of the 20th ISMIR Conf., ISMIR 2019*, pages 159–165, 2019.

[29] Liming Zhang, Wenbin Zhang, and Nathalie Japkowicz. Conditional-unet: A condition-aware deep model for coherent human activity recognition from wearables. In *2020 25th Int. Conf. on Pattern Pecognit. (ICPR)*, pages 5889–5896, 2021.

[30] Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. In *Proc. of the AAAI Conf. on Artificial Intelligence*, volume 32, 2018.

[31] Lu Lu, Ming Dao, Punit Kumar, Upadrasta Ramamurty, George Em Karniadakis, and Subra Suresh. Extraction of mechanical properties of materials through deep learning from instrumented indentation. *Proc. Natl. Acad. Sci.*, 117:7052–7062, 2020.

[32] Max Jaderberg, Karen Simonyan, Andrew Zisserman, and Koray Kavukcuoglu. Spatial transformer networks. *Adv. in Neural Inf. Proc. Syst.*, 2015-Janua:2017–2025, 2015.

[33] Xiaosong Du and Leifur Leifsson. Multifidelity model-assisted probability of detection via cokriging. *NDT & E Int.*, 108:102156, 2019.

[34] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In Nassir Navab, Joachim Hornegger, William M. Wells, and Alejandro F. Frangi, editors, *Med. Image Comput. and Comput.-Assisy. Interv. – MICCAI 2015*, pages 234–241, Cham, 2015. Springer Int. Publishing.

[35] Jana Fragemann, Lynton Ardizzone, Jan Egger, and Jens Kleesiek. Review of disentanglement approaches for medical applications – towards solving the gordian knot of generative models in healthcare. *arXiv e-prints*, 2022.

[36] Haris Iqbal. Harisiqbal88/plotneuralnet v1.0.0, December 2018.

[37] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. *Proc. IEEE Int. Conf. Comput. Vis.*, 2015 Inter:1026–1034, 2015.

[38] Dmitry Ulyanov, Andrea Vedaldi, and Victor S. Lempitsky. Instance normalization: The missing ingredient for fast stylization. *CoRR*, abs/1607.08022, 2016.

[39] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proc. IEEE Int. Conf. Comput. Vis.*, page 1510–1519, Venice, 09 2017. IEEE.

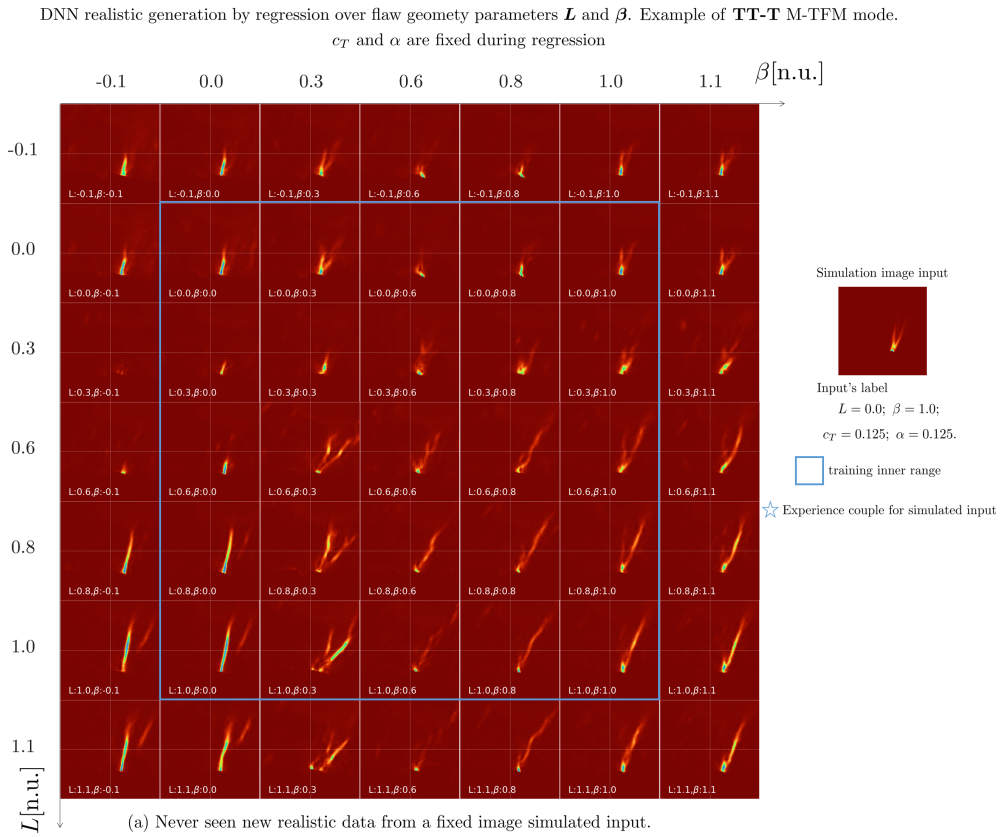
[40] Liming Jiang, Bo Dai, Wayne Wu, and Chen Change Loy. Focal Frequency Loss for Image Reconstruction and Synthesis. *Proc. IEEE Int. Conf. Comput. Vis.*, pages 13899–13909, 2021.

[41] Laurens Van Der Maaten and Geoffrey Hinton. Visualizing Data using t-SNE. *J. Mach. Learn. Res.*, 9:2579–2605, 2008.

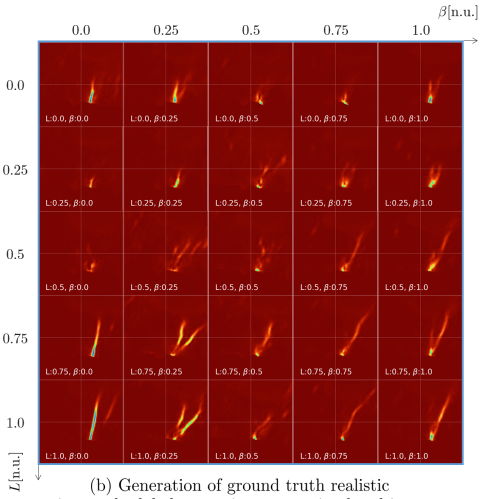
[42] Caifang Cai, Roberto Miorelli, Marc Lambert, Thomas Rodet, Dominique Lesselier, and Pierre-Emile Lhuillier. Metamodel-based markov-chain-monte-carlo parameter inversion applied in eddy current flaw characterization. *NDT & E International*, 99:13–22, 2018.

A Appendix: realistic data generation by trained tailored cU-Net.

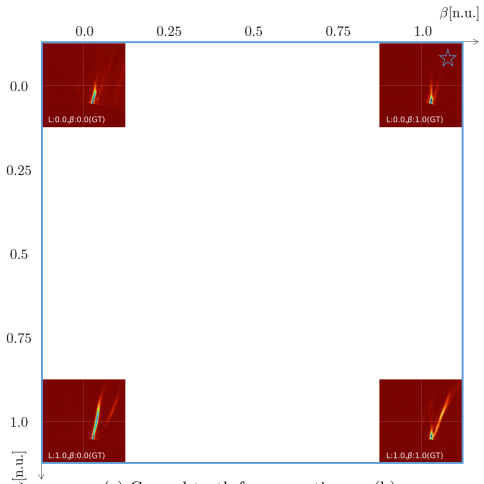
Figures demonstrate the DNN meta-model's generation potential for realistic M-TFM images. Comparison to ground truth is made where possible. Parameter variations $\boldsymbol{p}$ are employed for predictions. Consistency with physics is observed in terms of wave celerity and back-wall angle. However, accuracy decreases when altering flaw height and angle due to the limited diversity of the four-flaw experimental data set. Consequently, extrapolation performance cannot be deemed sufficiently accurate. Two reconstruction modes highlight distinct impacts of flaw geometry and reconstruction parameters.



(a) Never seen new realistic data from a fixed image simulated input.

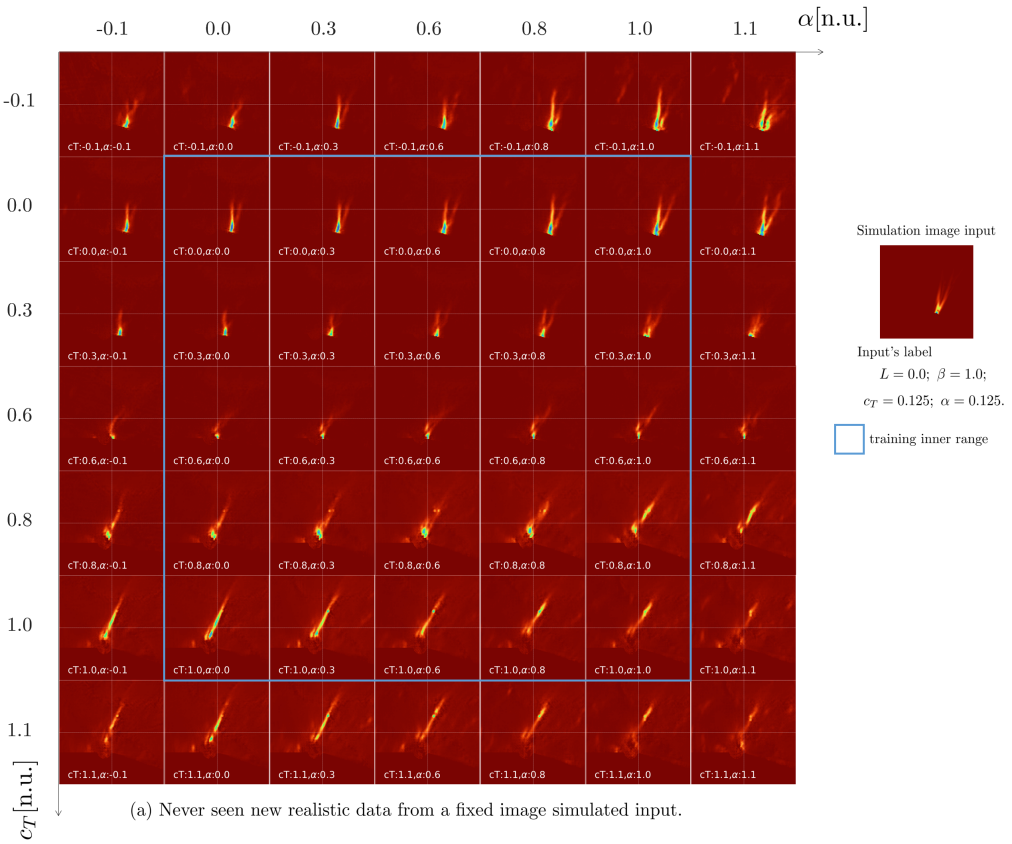


(b) Generation of ground truth realistic images by label regression over a simulated input.

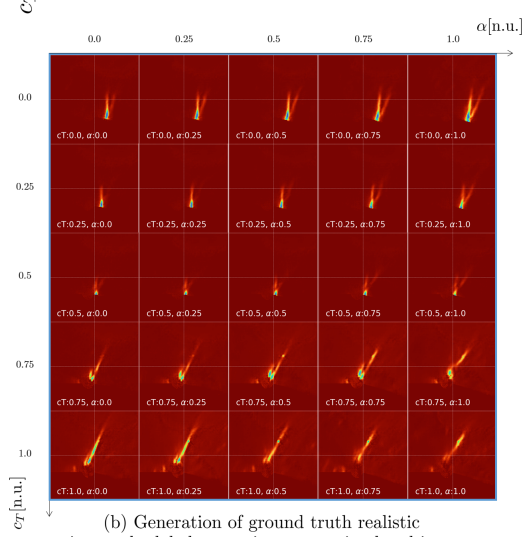


(c) Ground truth for generation on (b).

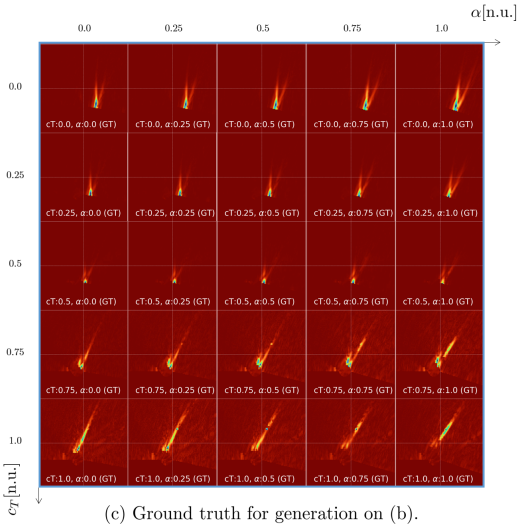
DNN realistic generation by regression over M-TFM reconstruction parameters $\boldsymbol{c_T}$ and $\boldsymbol{\alpha}$. Example of **TT-T** M-TFM mode.
L and β are fixed during regression



(a) Never seen new realistic data from a fixed image simulated input.

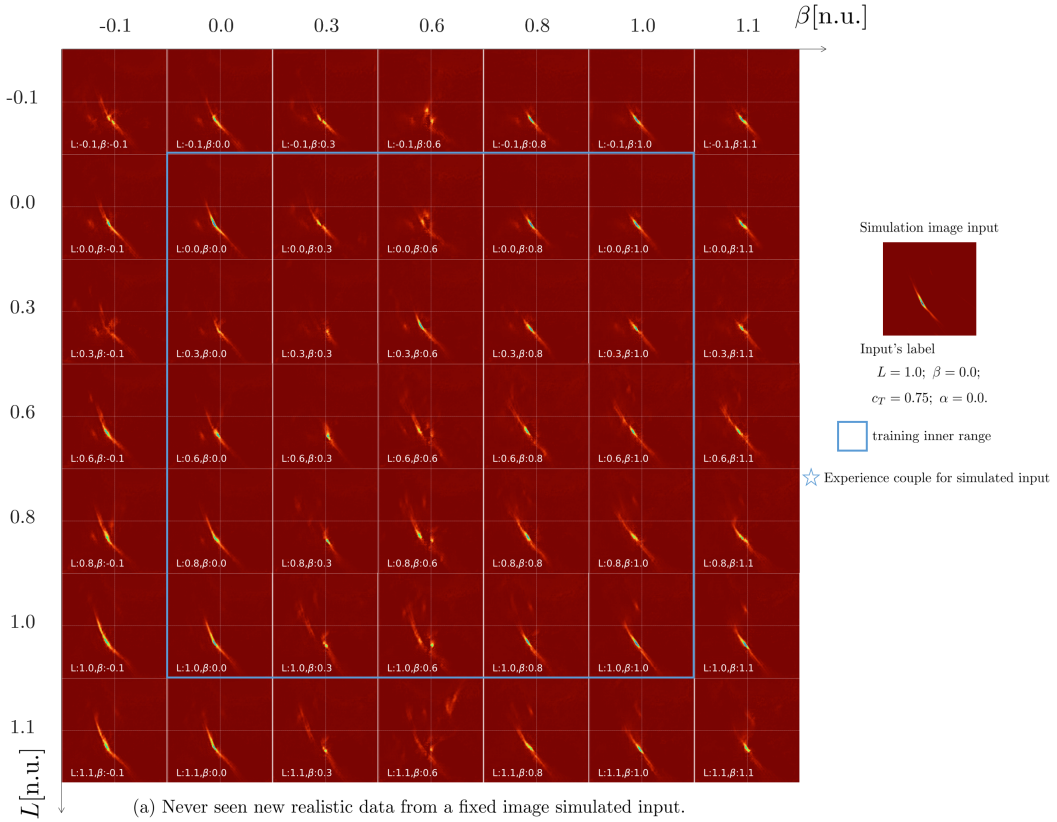


(b) Generation of ground truth realistic images by label regression over a simulated input.

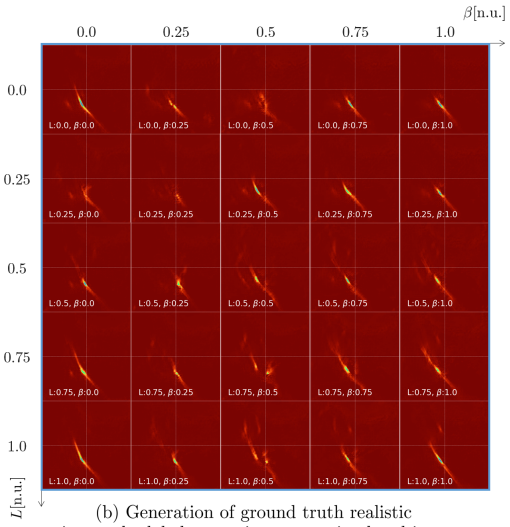


(c) Ground truth for generation on (b).

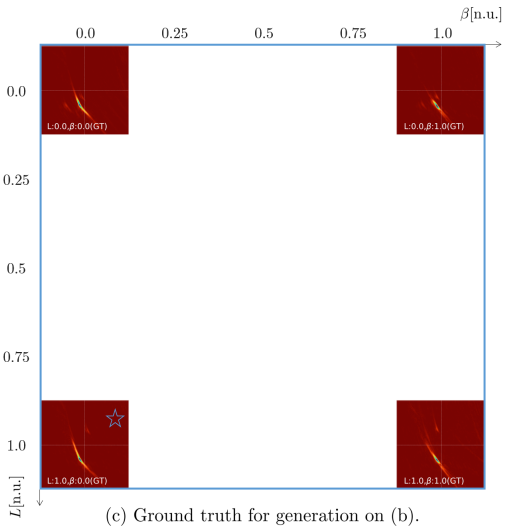
DNN realistic generation by regression over flaw geomety parameters $\boldsymbol{L}$ and $\boldsymbol{\beta}$. Example of **TT** M-TFM mode.
c_T and α are fixed during regression



(a) Never seen new realistic data from a fixed image simulated input.

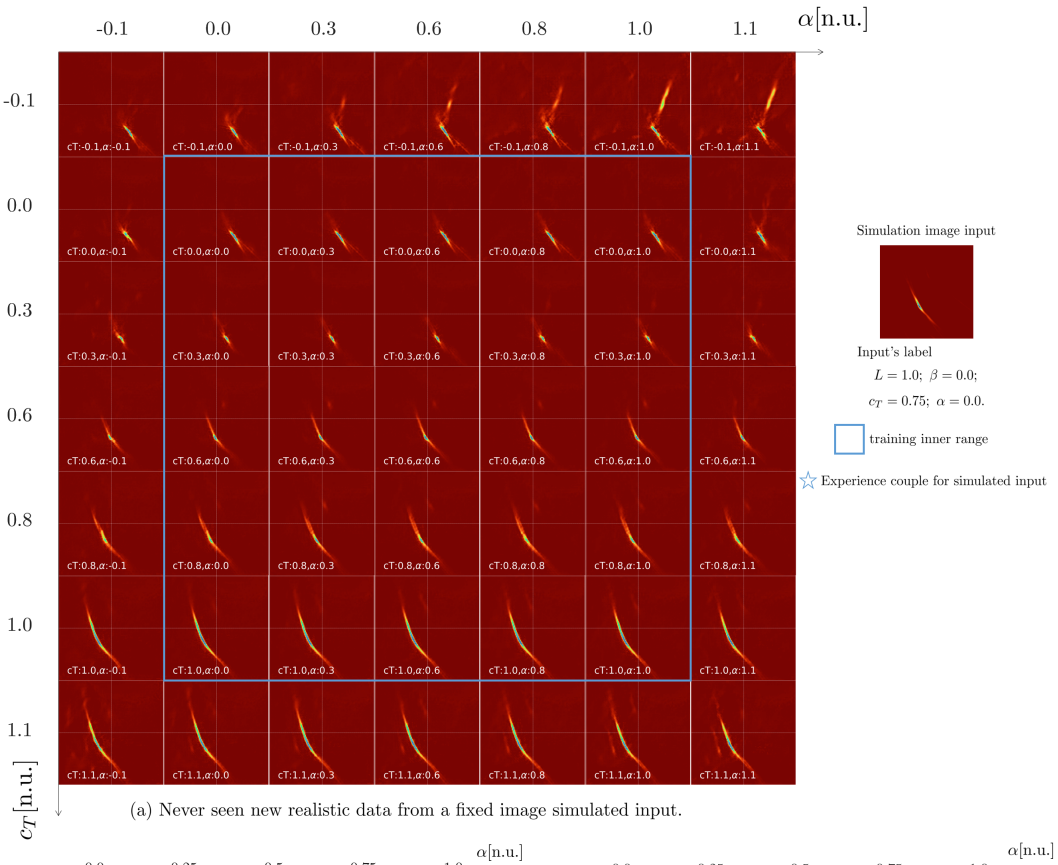


(b) Generation of ground truth realistic images by label regression over a simulated input.

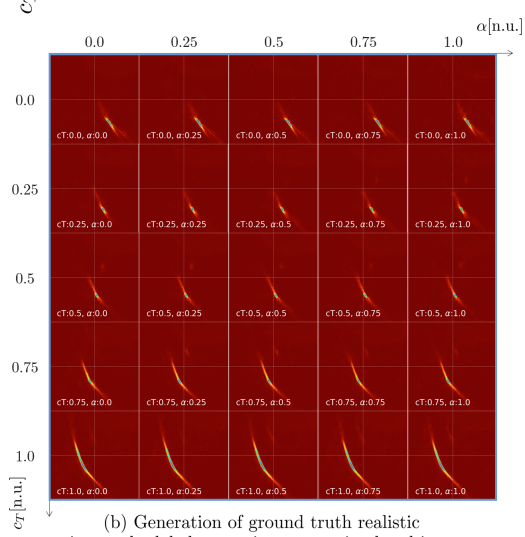


(c) Ground truth for generation on (b).

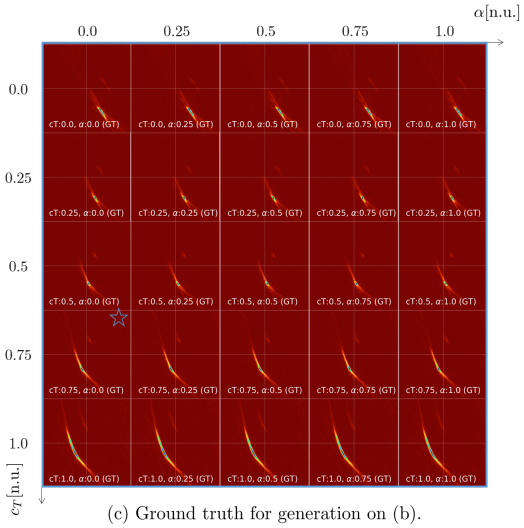
DNN realistic generation by regression over M-TFM reconstruction parameters $\boldsymbol{c_T}$ and $\boldsymbol{\alpha}$. Example of **TT** M-TFM mode.
L and β are fixed during regression



(a) Never seen new realistic data from a fixed image simulated input.



(b) Generation of ground truth realistic images by label regression over a simulated input.



(c) Ground truth for generation on (b).