



A deep learning model to predict the failure response of steel pipes under pitting corrosion

Mingshi Ji¹ · Ming Yang¹ · Soheil Soghrati^{1,2}

Received: 21 July 2022 / Accepted: 12 October 2022 / Published online: 29 October 2022
© The Author(s), under exclusive licence to Springer-Verlag GmbH Germany, part of Springer Nature 2022

Abstract

Pitting corrosion is one of the major causes of failure in high-pressure oil and gas pipelines. Various inspection techniques can be used to characterize the morphology of corrosion pits, which must be linked to the risk of failure to develop proper maintenance strategies. While numerical techniques such as the finite element method can accurately predict this risk, the labor and computational cost associated with these methods render their application unfeasible over hundreds of miles of a pipeline. In this manuscript, we introduce a deep learning approach relying on the squeeze-and-excitation residual network (SE-ResNet) to predict the strength and toughness of statistical volume elements (SVEs) of a corroded pipe. An automated microstructure reconstruction and mesh generation framework is utilized to synthesize the training data for this model by simulating the failure response of 10,000 SVEs subject to a tensile load (hoop stress). A Bayesian optimization approach is utilized to determine the optimal combination of hyperparameters for the SE-ResNet model, followed by a k -fold cross-validation of the model. We show that the trained SE-ResNet can accurately predict the failure response of corroded pipe SVEs with a maximum error of $< 1\%$. Moreover, a comparison between the proposed model with several other well-known DL architectures shows that it yields superior accuracy and efficiency.

Keywords Deep learning · Pipe · Pitting corrosion · Squeeze-and-excitation residual network · Finite element method

1 Introduction

Steel pipes are widely used in underground networks for oil and gas transportation. This exposes the pressurized pipes to a corrosive environment, which leads to various types of corrosion attacks and in particular the pitting corrosion [1]. Pitting corrosion is a localized form of corrosion that can penetrate the mass of the metal with a limited diffusion of ions [2]. Corrosion pits could cause a significant reduction in the structural integrity of pressurized pipes and eventually lead to the pipe leakage or fracture. The failure of buried pipes for oil and gas transportation could have catastrophic economic and environmental effects. Therefore, it is essential to study the corrosion-assisted failure response of pipes and

develop proper failure prediction tools to improve the safety and facilitate their maintenance.

According to safety regulations, an internal inspection must occasionally be conducted on oil/gas pipelines to identify the mass loss and defects due to phenomena such as pitting corrosion. Various diagnostic techniques can be used for inline inspection of a pipeline, which enables collecting valuable data (over tens to hundreds of miles of lengths) by recording its internal surface morphology [3]. Conventional inline inspection instruments (pigs) are properly sized spherical or cylindrical devices that are propelled through the interior of a pipeline by adjusting the pressure and flow, or by mechanically pushing the device through the pipeline [4]. These conventional pigs are always designed for specific pipelines instead of applying to different pipelines. However, the pipeline might be “unpiggable” if the inspection instrument cannot physically pass through due to conditions such as changing pipeline diameter and flow limits [5]. To satisfy integrity management rules, smart pigging has taken a more dominant role in the past few years. Smart pigs are a collection of complex electronic components including computers for receiving and storing data, sensors for detecting different

✉ Soheil Soghrati
soghrati.1@osu.edu

¹ Department of Mechanical and Aerospace Engineering, The Ohio State University, 201 West 19th Avenue, Columbus, OH, USA

² Department of Materials Science and Engineering, The Ohio State University, 201 W. 19th Avenue, Columbus, OH, USA

irregularities, and power sources for operating these electronics. Magnetic flux leakage (MFL) and ultrasonics (UT) are two major technologies used by smart pigs to obtain information regarding the interior condition of pipelines [6]. MFL inspects pipelines by injecting magnetic flux into the pipe walls to identify leaks, corrosion, and defects. Ultrasonic inspection directly estimates the pipe's wall thickness by employing ultrasonic waves to measure the time it takes for an echo to back to the sensor [7].

The piping system is subjected to various mechanical loads caused by internal pressure, weight, and temperature changes during its service life. High stress concentrations developed near corrosion pits could significantly accelerate the failure and lead to pipe leaks and bursts. Determining the strength of corroded pressurized pipes is closely related to the failure response of plates impacted by pitting corrosion. A significant volume of research is focused on studying the effect of corrosion pits on the failure response of metallic structures, including pressurized pipes. For example, the smallest cross-sectional area is proposed as a damage parameter to describe the reduction of ultimate strength for steel plates subjected to pitting corrosion under axial tensile and compressive loads [8]. The relationship between the ultimate shear strength of a pitted plate and the degree of pitting corrosion intensity has been studied in [9]. An alternative method for assessing the ultimate strength of plates under uniaxial compression is to use the corroded volume loss as the measurement of the pitting corrosion damage [10].

Several studies have used analytical techniques to predict the failure response of metallic structures with corrosion pits. For example, an analytical approach is introduced in [11] to calculate the failure strength of corroded steel plates subjected to quasi-static tensile loading. The failure of corroded pipes is caused by the transition of pits to cracks, followed by crack propagation. The pit-to-crack transition is commonly modeled using two methods based on fracture mechanics [12–15] or using a continuum damage model (CDM) [16–18]. In fracture mechanics based approaches, once the stress intensity factor around pits reaches the threshold of crack growth, a crack nucleates from one pit [19]. The critical stress intensity factor can be calculated according to the pit shape and size [14]. In CDM-based approaches, the pit is regarded as a notch and the evolution of pits is assumed to follow Faraday's law [16,20].

A higher fidelity approach to quantify the effect of corrosion pits on the mechanical integrity is to implement numerical techniques such as the finite element method (FEM) to approximate the failure response of the pipe [1,21–23]. Several studies have shown the ability of FEM to accurately predict the failure response of corroded pipes when appropriate constitutive models are employed [24–26]. FEM is especially beneficial for predicting the failure behavior of pipelines with varying corrosion pit shapes/sizes.

However, when implemented over a large set of data similar to corroded pipe surface morphology images collected during an inline inspection over hundreds of miles of a pipeline length, this process would be exceedingly laborious and computationally expensive. In other words, the time-consuming process associated with converting imaging data into geometrical models (e.g., CAD files), generating thousands of FE meshes, and performing corresponding nonlinear failure simulations makes it practically impossible to use FEM for the assessment of the risk of failure in pipelines.

Given the prior success of deep learning (DL) algorithms in applications involving computer-aided diagnosis and predictions [27–29], developing/training a DL model to predict the pipe failure response directly based on its surface images could be a viable alternative. To enumerate a handful of works in this field, we can mention the artificial neural network and the multi-variable regression approach in [30,31] to predict the ultimate strength of unstiffened plates with localized corrosion. A 3D convolutional neural network (CNN) model was implemented by Yang et al. [32] to predict the effective stiffness of high contrast elastic composites. Another 3D CNN model was trained to predict the anisotropic effective properties of particle reinforced composites in [33]. The plastic behavior of composite representative volume elements (RVEs) was successfully predicted using Recurrent Neural Networks (RNNs) in [34]. Moreover, Physics Informed Neural Networks (PINNs) are introduced for predicting the linear elastic and nonlinear elastoplastic of materials [35].

Developing an efficient and accurate DL model could be a challenging and time-consuming task, which in addition to selecting an appropriate algorithm, requires optimizing the model parameters and hyperparameters [36]. The former group must be determined based on the training data set, while the latter is tuned before the training process to achieve the optimal model architecture and the best performance [37,38]. While the manual tuning of hyperparameters is widely used for most applications, it is an inefficient approach for complex models with a large number of hyperparameters and requires an in-depth knowledge of the DL model by the user [39]. To overcome these challenges, several hyperparameter optimization techniques (HPO) are developed to automate the tuning process [40–44]. Grid search (GS) is one of the earliest techniques, where hyperparameters evolve into an optimal combination after training the model and evaluating its performance [43]. However, the GS method could be highly computationally demanding, especially for complex model architectures trained over a massive set of data. An alternative approach is a random search (RS) method, which generates random samples with random parameters from a given statistical distribution [45]. Bayesian optimization has also been applied to hyperparameter tuning [46], which often requires fewer iterations compared to the RS technique, as it determines the optimal combination of hyperparameter val-

ues since it selects parameter combinations in an informed way.

In this manuscript, we present a DL model, trained using high fidelity FE simulation results, to predict the strength and toughness of statistical volume elements (SVEs) of a steel pipe subjected to pitting corrosion. Note that currently this model is built for one type of steel (X100) typically used in high-pressure pipes, and therefore only receives geometric shapes (images) of different corroded pipe SVEs as the input. An automated computational framework is employed to synthesize more than 10,000 corroded metal SVEs with distinct shapes, sizes, and a spatial arrangement of corrosion pits, generate 3D conforming meshes, and perform high-fidelity failure FE analyses (under a tensile load) to evaluate their strength and toughness. Gray-scale images of corroded metal SVEs, labeled with resulting strength/toughness values, were then used to train a squeeze-and-excitation Residual Network (SE-ResNet) to predict these values directly from the imaging data. The Bayesian optimization approach is employed to determine the optimal combination of hyperparameters for this SE-ResNet model. After the k -fold cross-validation, we show that this model can predict the failure strength and toughness of corroded pipelines with high accuracy (maximum error: $< 1\%$) and at a fraction of the computational and labors costs associated with FE simulations (≈ 5 orders of magnitude speedup).

The remainder of this manuscript is structured as follows. In Sect. 2, we present the governing equations and the continuum damage model used for approximating the failure response of corroded steel pipe SVEs. The automated computational framework used for the virtual reconstruction and FE meshing of these SVEs, together with the image processing based approach employed for building the training data set, is presented in Sect. 3. The SE-ResNet model architecture, evaluation of its optimal hyperparameters, and the k -fold validation of the model predictions are presented in Sect. 4. Final concluding remarks are summarized in Sect. 5.

2 Problem formulation

2.1 Governing equations

As noted previously, the training data for the SE-ResNet model used in this work are acquired through the high-fidelity FE simulation of the failure response of steel pipe SVEs. Let Ω be the macroscopic domain of this pipe characterized in the coordinate system $\mathbf{x}_M$, while Θ refers to the microscopic domain of each SVE defined in the coordinate system $\mathbf{x}_m$. The SVE boundaries are shown by Γ , which have a unit normal vector $\mathbf{n}_m$.

Using the first order asymptotic expansion, we can decompose the displacement field $\mathbf{u}(\mathbf{x}_M, \mathbf{x}_m)$ into its macroscopic

$\mathbf{u}_M(\mathbf{x}_M)$ and microscopic $\mathbf{u}_m(\mathbf{x}_M, \mathbf{x}_m)$ components as

$$\mathbf{u}(\mathbf{x}_M, \mathbf{x}_m) = \mathbf{u}_M(\mathbf{x}_M) + \xi \mathbf{u}_m(\mathbf{x}_M, \mathbf{x}_m). \quad (1)$$

Similarly, the strain field can be broken down into macroscopic and microscopic components, i.e., $\boldsymbol{\varepsilon}_M$ and $\boldsymbol{\varepsilon}_m$, respectively. The linear elasticity governing equations at the microscale (for each SVE) can be written as

$$\nabla \mathbb{C} : (\boldsymbol{\varepsilon}_M + \boldsymbol{\varepsilon}_m) = \mathbf{0} \quad \text{in } \Theta, \quad (2)$$

where $\mathbb{C}$ is the fourth-order elasticity tensor.

To calculate the macroscopic energy density Φ_M at a given point in the macroscopic domain, one can implement the Hill-Mandel micro-homogeneity principle [47] given by

$$\inf_{\mathbf{u}_M} \Phi_M(\boldsymbol{\varepsilon}_M) = \inf_{\boldsymbol{\varepsilon}_m} \inf_{\mathbf{u}_m} \frac{1}{|\Theta|} \int_{\Theta} \Phi_m(\boldsymbol{\varepsilon}_M + \boldsymbol{\varepsilon}_m) d\Theta. \quad (3)$$

where Φ_m is the average energy density of the SVE corresponding to the macroscopic point. Φ_M and Φ_m are calculated as

$$\Phi_M = \frac{1}{2} \boldsymbol{\varepsilon}_M : \boldsymbol{\sigma}_M, \quad \Phi_m = \frac{1}{2} (\boldsymbol{\varepsilon}_M + \boldsymbol{\varepsilon}_m) : \boldsymbol{\sigma}_m, \quad (4)$$

where $\boldsymbol{\sigma}_M$ and $\boldsymbol{\sigma}_m$ are macroscopic and microscopic stress tensors, respectively. A homogenization approach can then be utilized to evaluate the macroscopic stress tensor as

$$\boldsymbol{\sigma}_M(\mathbf{x}_M) = \frac{1}{|\Theta|} \int_{\Theta} \boldsymbol{\sigma}_m(\mathbf{x}_m) d\Theta. \quad (5)$$

A similar concept is used in the strain averaging theorem to relate the macroscopic strain tensor at a given point to the average of microscopic strains in its corresponding SVE as

$$\boldsymbol{\varepsilon}(\mathbf{x}_M) = \frac{1}{|\Theta|} \int_{\Theta} \boldsymbol{\varepsilon}(\mathbf{x}_m) d\Theta. \quad (6)$$

The governing equations above can be used to evaluate the effective properties (e.g., strength and toughness) of the pipe at a given point based on the FE approximation of the mechanical response of the corresponding SVE. To achieve this, we must also consider the elastoplastic behavior and the development/propagation of cracks in the steel. In other words, instead of using the elasticity tensor corresponding to a linear elastic behavior in (2), we must implement an appropriate constitutive model to take into account the material nonlinear behavior. To achieve this, we use a phenomenological continuum ductile damage [48] model, as described next.

2.2 Constitutive damage model

The plastic strain accumulated in each SVE under mechanical loading would eventually lead to the development of micro-cracks and micro-void coalescence. In the constitutive model used here [48], the effect of damage at each point is modeled as

$$D = 1 - \frac{A_D}{A_0}, \quad (7)$$

where $0 \leq D \leq 1$ is a scalar damage parameter reflecting the intensity of damage, while A_0 and A_D are cross-section areas of the undamaged and damaged surfaces, respectively. Note that $D = 0$ and $D = 1$ denote the intact and fully damaged states of the material at a given point, respectively. In the context of FEM, D reflects the state of damage in each element, meaning an element reaching the fully damaged state ($D = 1$) has 0 stiffness, which will be deleted from the mesh during the FE simulation.

The initial (intact state) modulus of elasticity, E_0 , of the material after the damage becomes

$$E_D = E_0(1 - D), \quad (8)$$

where E_D is the effective modulus of the damaged material. The strain energy release rate Y can then be expressed as [49]

$$Y = \frac{\sigma_{eq}^2}{2E(1-D)^2} \left[\frac{2}{3}(1+\nu) + 3(1-2\nu) \left(\frac{\sigma_m}{\sigma_{eq}} \right)^2 \right], \quad (9)$$

where σ_m is the hydrostatic stress and $\sigma_{eq} = \left(\frac{3}{2} s_{ij} s_{ij} \right)^{1/2}$ is the equivalent von Mises stress (s_{ij} : deviatoric stress tensor). The elastic strain rate tensor is given by

$$\dot{\epsilon}^e = \frac{1+\nu}{E} \frac{\dot{\sigma}}{1-D} - \frac{\nu}{E} \frac{\dot{\sigma}_t}{1-D} \mathbf{I}, \quad (10)$$

where σ_t is the trace of the stress tensor and $\mathbf{I}$ is the Kronecker delta tensor. Under the isothermal assumption, the plastic strain rate $\dot{\epsilon}^p$ can be evaluated as

$$\dot{\epsilon}^p = \lambda \Psi = \lambda \frac{\partial F_p}{\partial \sigma}, \quad (11)$$

where λ is the plastic multiplier and Ψ is the direction of the strain increment derived from the plastic potential F_p .

The yield surface is defined as

$$f(\sigma) = q - \sigma_Y(\epsilon_{eq}^p), \quad (12)$$

where ϵ_{eq}^p is the equivalent plastic strain and $\sigma_Y(\epsilon_{eq}^p)$ is the yield function determined based on an experimental stress–strain curve. The damage initiates when ϵ_{eq}^p reaches

a threshold value, ϵ_0^p , which is a function of the equivalent plastic strain rate $\dot{\epsilon}_{eq}^p$ and the stress triaxiality η . A state variable S_{in} is introduced to predict the initiation of damage when $S_{in} = 1$, which is defined as

$$S_{in} = \int \frac{d\epsilon_{eq}^p}{\epsilon_0^p}(\eta, \dot{\epsilon}_{eq}^p). \quad (13)$$

After damage initiation, the damage parameter D monotonically increases from 0 to 1, causing the degradation of material stiffness. To avoid the ill-conditioning associated with this phenomenon, any element for which D reaches 1 is deleted from the mesh. To reduce the mesh dependency effects during the simulation, we use a non-local regularization approach, where an effective plastic displacement u^p is used in approximating the damage initiation/evolution based on the characteristic length factor L . The rate of u^p is given by [50]

$$\dot{u}^p = L \dot{\epsilon}_{eq}^p. \quad (14)$$

In this approach, it is assumed that the damage evolution parameter exponentially varies with u^p . The damage parameters used in this work are calibrated based on the experimental data reported in [51,52] (X100 steel properties are provided in Table 1). More details regarding this damage model, its FE implementation, and calibration procedure are presented in [53–55].

3 Acquiring training data

To acquire the training data for the SE-ResNet model, we implement an automated computational framework to (a) synthesize more than 10,000 corroded pipe SVEs, (b) generate high-fidelity FE models, and (c) simulate the failure response of each SVE under a tensile load to approximate its strength and toughness. Each step of this process is described in more detail next.

3.1 Virtual microstructure reconstruction

The shape, size, and spatial arrangement of corrosion pits have a strong impact on local mechanical properties of the pipe, which necessitates realistically incorporating them in each SVE. As noted previously, ultrasonic inline inspection techniques can readily capture gray-scale images of the pipe surface, which can then be converted to 3D geometrical SVE models. In the absence of such proprietary data for this study, here we implement a virtual microstructure reconstruction algorithm [56] to synthesize thousands of realistic corroded pipe SVEs and acquire the training data. In this approach, a library of corrosion pit morphologies (characterized in terms

Table 1 Material properties of X100 steel used in this paper

Elastic modulus (GPa)	Poisson ratio	Yield strength (MPa)	Tensile strength (MPa)	Displacement at failure (mm)
210	0.3	710	990	0.1

of NURBS) are virtually packed in the SVE domain using a set of hierarchical bounding box (BBox) to avoid overlaps between inserted pits (see [56] for more details). Before starting the packing process, all virtual pits in the shape library are sorted in a descending order by size (volume). A pit is then randomly selected from the shape library, rotated at an arbitrary angle, and added to the SVE at a random location provided that the BBox-based algorithm does not indicate an overlap with previously added pits. Otherwise, a new random location is selected and the packing process is recursively continued until reaching the desired volume fraction. In this work, we virtually reconstructed 10,000 SVEs with a pit volume fraction ranging from 0.1 to 4.0%, six of which are illustrated in Fig. 1.

It is worth mentioning that although all the SVEs studied in this work are virtually reconstructed, even in the presence of actual surface morphology data (e.g., gray-scale images collected via ultrasonic pigging), this approach might still be required to supplement the training data. For example, assume a case a scenario that along the pipe length we could rarely see SVEs with a pit volume fraction of $> 5\%$. While such heavily corroded regions are detrimental to the structural integrity of the pipe, their scarcity may complicate the training process, leading to a low accuracy in predicting the strength/toughness of SVEs. A virtual microstructure reconstruction approach can be implemented to enrich the training data by synthesizing SVEs with statistically equivalent microstructures in cases that the original data set lacks the desired diversity.

3.2 CISAMR Mesh generation

To build the training data, i.e., the SVE microstructure labeled with the corresponding strength/toughness, all 10,000 virtually reconstructed SVEs must be transformed into high-fidelity FE models to simulate their failure response. Such massive mesh generation effort requires utter automation of the modeling process, which in this work is accomplished using the Conforming to Interface Structured Adaptive Mesh Refinement (CISAMR) [57,58] technique. CISAMR is a non-iterative meshing algorithm that transforms a structured tetrahedral mesh overlaid with the domain geometry into a conforming mesh assuring that resulting element aspect ratios do not exceed 5. This mesh transforming process involves four major steps, namely the h -refinement and r -adaptivity of background elements near material interfaces, face-swapping of distorted elements, and sub-

tetrahedralization to build the final conforming mesh. See [58] for more details regarding the CISAMR algorithm and implementation for modeling 3D problems.

To select an appropriate mesh size that ensures the failure response of SVEs is not mesh-dependent, we performed a mesh convergence study, which is reported in Table 2 for one SVE. According to this study, a $100 \times 100 \times 10$ background mesh with one level of h -adaptive refinement along corrosion pit surfaces was adopted to generate FE models of corroded SVEs using CISAMR to perform the simulations needed for training the DL model. Figure 2 illustrates the resulting conforming mesh for one of these SVEs, which is composed of 6.2×10^5 elements. As shown in the inset of this figure, CISAMR yields high-quality conforming elements with proper aspect ratios (maximum of 3.9) and a negligible geometric discretization error.

3.3 FE failure simulations

After generating FE models of all 10,000 SVEs, an explicit solver is employed to simulate their failure response subject to a tensile load and approximate their strength/toughness using the governing equations described in Sect. 2. To build the training data, we must then evaluate the strength and toughness of each SVE. The failure toughness (T) is defined as the energy of mechanical deformation per unit volume before fracture, which is calculated by integrating the stress–strain curve as

$$T = \int_0^{\varepsilon_f} \sigma \, d\varepsilon, \quad (15)$$

where ε_f is the failure strain. The simulated damage patterns in three of these SVEs at the initial stages of damage nucleation and after failure are illustrated in Fig. 3. As expected, the corrosion pits play a crucial role in the initiation of damage in each SVE, and depending on their shape/size, some of these pits are detrimental to the formation of major cracks leading to failure (pipe rupture). Figure 4 shows the resulting homogenized stress–strain response of 1000 (10%) of SVEs, indicating a large variation in their strength (maximum sustained stress before failure) and toughness (the area under each curve). Figure 5a better shows distributions of failure strength and toughness approximated using high-fidelity FE simulations, with the former varying from 878.8 to 990.1 MPa and the latter from 62.7 to 238.4 MJ/mm³. Note that,



Fig. 1 Six virtually constructed SVEs of the corroded pipe with various shapes, sizes, spatial arrangements, and volume fractions of corrosion pits

Table 2 Predicted values of an SVE strength for 5 different FE models generated using CISAMR on different background meshes

	Mesh 1	Mesh 2	Mesh 3	Mesh 4	Mesh 5
Background mesh	$40 \times 40 \times 4$	$60 \times 60 \times 6$	$80 \times 80 \times 8$	$100 \times 100 \times 10$	$150 \times 150 \times 15$
Number of elements	5.2×10^4	1.55×10^5	3.21×10^5	6.23×10^5	1.86×10^6
Strength (MPa)	929.6	918.4	915.7	915.2	915.2

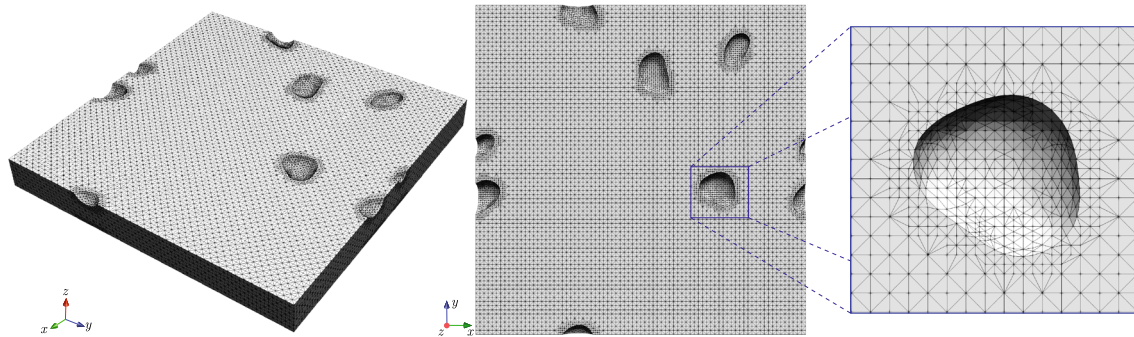


Fig. 2 Conforming mesh generated using CISAMR for one of the virtually reconstructed SVEs of the steel pipe

for example, the failure strength of the SVE with pit volume fraction 4% reduces by 11.4% compared to an intact SVE.

hyper-parameters of this SE-ResNet model, followed by a k -fold cross-validation to examine its performance on new data (SVEs not seen during the training process).

4 DL model architecture and training

In this section, we describe how the FE simulation results are used to train a DL model (SE-ResNet) to directly predict the strength and toughness of each SVE from a gray-scale image of the corroded metal surface. The Bayesian optimization approach is implemented to determine optimal

4.1 Building the training data set

As noted previously, simulated strength and toughness of 10,000 corroded pipe SVEs are utilized for training the SE-ResNet model. While each SVE (and the corresponding FE model) has a 3D geometry, input images to this DL model can be characterized as 2D grayscale images of the corroded SVE

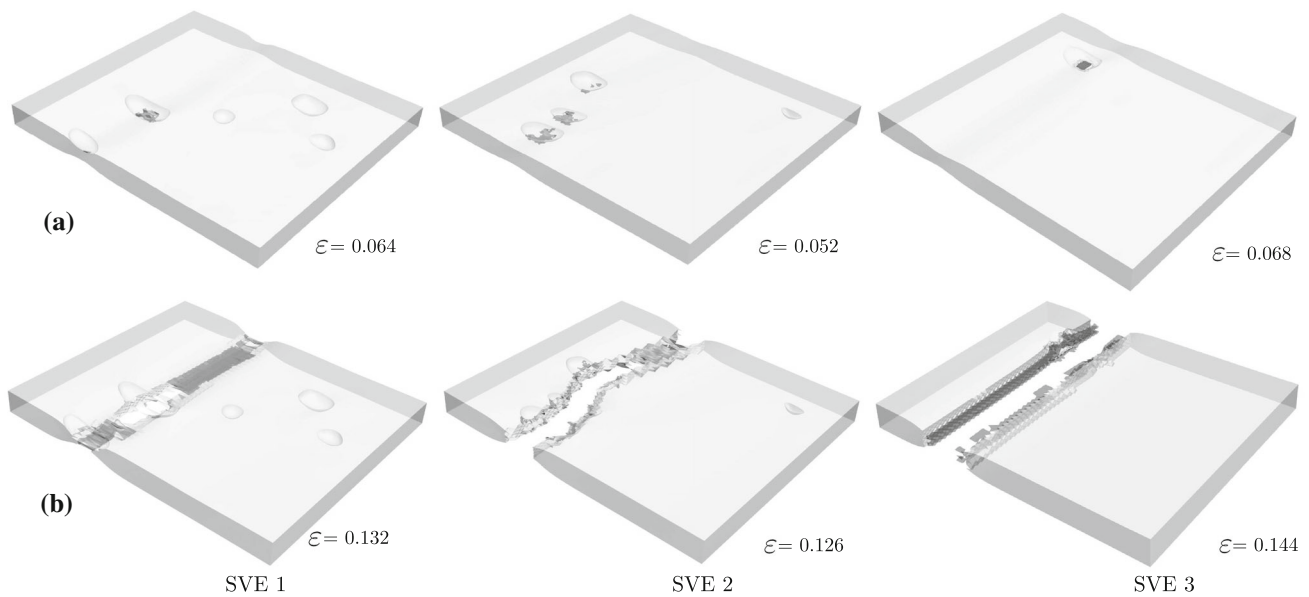
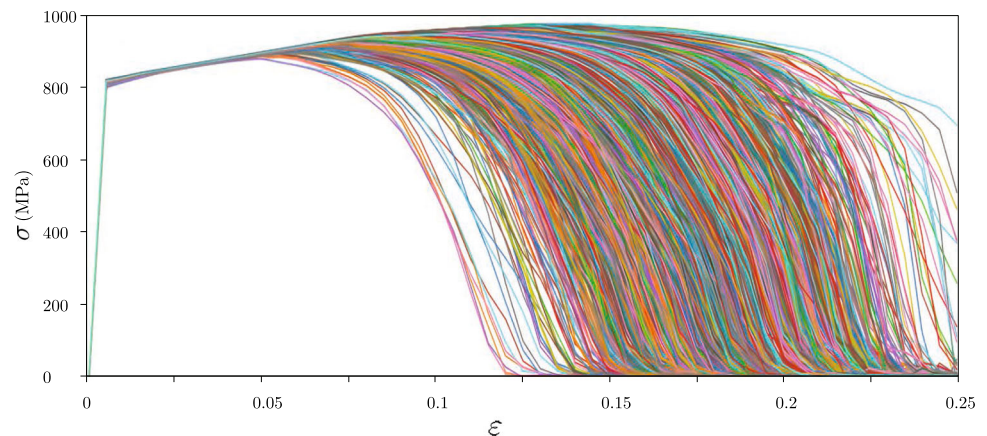


Fig. 3 Simulated damage patterns **a** in early stages of damage nucleation and **b** after failure in 3 SVEs subjected to a tensile load

Fig. 4 Stress–strain responses of 1000 samples under tensile load



surface. The grayscale intensity (ranging from 0 to 255) in these images represents the pit depth, with 0 being the intact surface of the SVE. To generate these images, the depth h of each point on the corroded surface (xy plane) is transformed to a scalar value $0 \leq p \leq 255$ as

$$p = \left\lfloor 255 \left(1 - \frac{h}{H} \right) \right\rfloor, \quad (16)$$

where H is the SVE thickness. Figure 6 shows two corroded SVEs and their corresponding grayscale images generated using this approach. Note that the images generated after this transformation preserve all geometric features of corrosion pits in the 3D model.

It is well known that increasing the size/diversity of training data often leads to a better predictive capability of the DL model. In most cases, the variance and bias also show a similar decrease when using a larger training dataset [59]. In

this work, increasing the size of the training set requires performing more high-fidelity FE simulations, which could be a computational task. Note that the number of simulations carried out to build the training data (10,000) has already been determined after a few attempts at training the DL model using smaller data sets, followed by increasing the number (performing more simulations) to achieve acceptable accuracy. However, a simple data augmentation strategy can be used to artificially increase the size of training data by 4 times at practically no computational cost. Unlike traditional classification models, which are invariant to the translation, viewpoint, and size of input images, the failure response of corroded SVEs is highly dependent on the spatial arrangement of pits. However, under a tensile load in the x -direction, the SVE strength and toughness remain intact if it is flipped along x or y axis, as well as being rotated by 180° , as shown in Fig. 7. Through this simple data augmentation strategy,

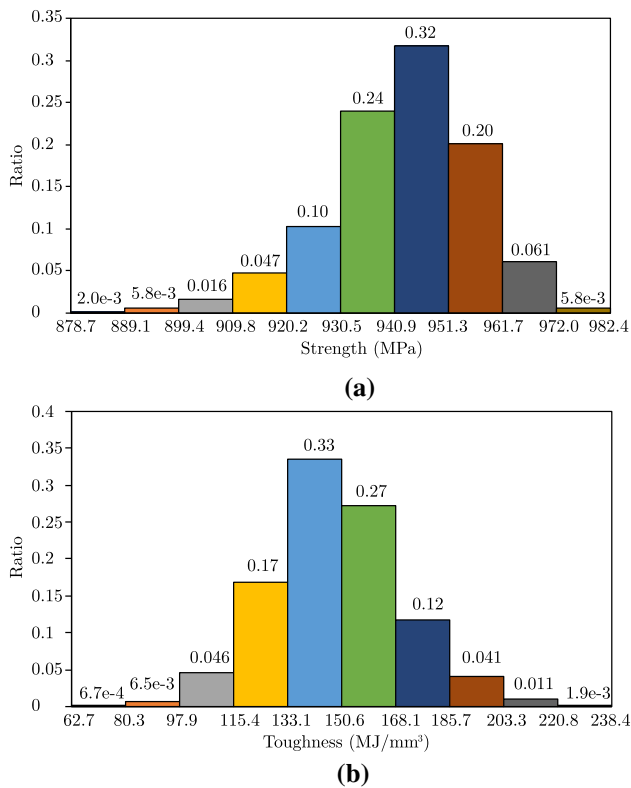


Fig. 5 Columnar distributions of **a** failure strength and **b** toughness of all SVEs under a tensile load approximated via high-fidelity FE simulations

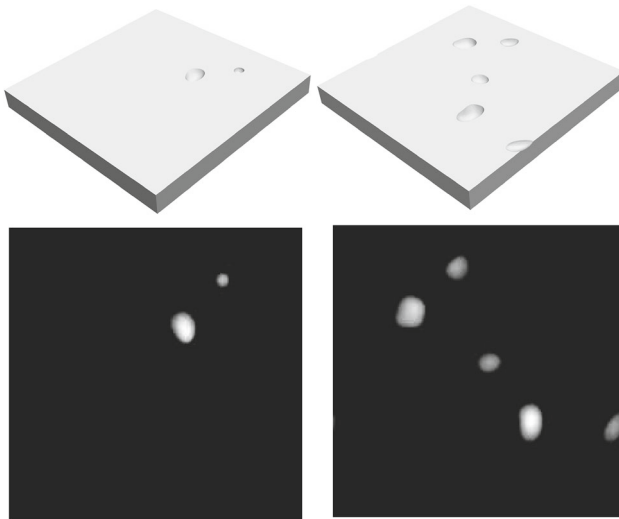


Fig. 6 Transformation of 3D corroded SVE microstructures into 2D grayscale images, which serve as input images to the SE-ResNet

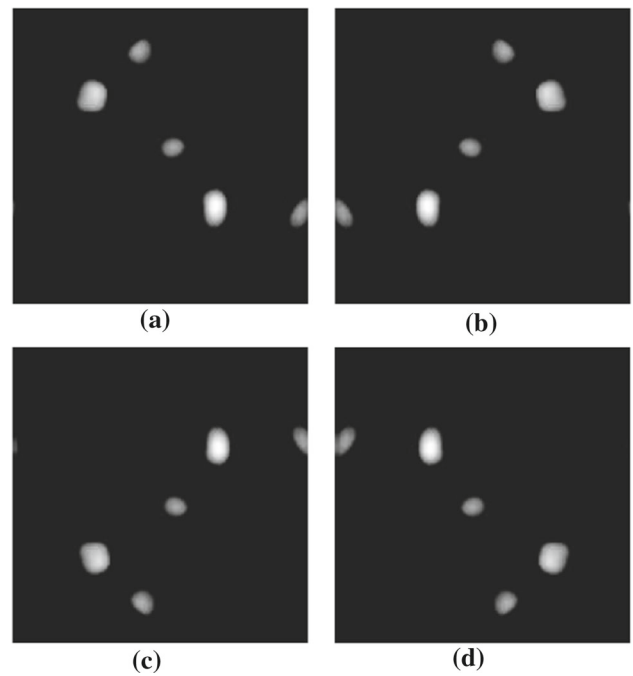


Fig. 7 Data augmentation: **a** original gray-scale image of an SVE; **b** mirrored along vertical axis; **c** mirrored along horizontal axis; **d** rotated by 180°

we have increased the size of labeled data used for training/validation of the SE-ResNet model to 40,000.

4.2 Proposed deep learning architecture

The SE-Net architecture was first introduced in [60] for image classification, which can significantly improve the performance of traditional CNNs with only a small increase in the computational cost. The squeeze-and-excitation (SE) block used in this model adaptively recalibrates channel-wise feature responses by explicitly modeling the channel interdependence. A transformation $\mathbf{P}_{tr}$ can be implemented to build the SE block by mapping an input $\mathbf{X} \in \mathbf{R}^{H' \times W' \times C'}$ to an output $\mathbf{U} \in \mathbf{R}^{H \times W \times C}$. The transformation $\mathbf{P}_{tr}$ is considered as a convolutional operator composed of a set of filter kernels $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n]$. The output $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_n]$ can be written as

$$\mathbf{u}_n = \mathbf{v}_n * \mathbf{X} = \sum_{s=1}^n \mathbf{v}_n * \mathbf{x}_s, \quad (17)$$

where $\mathbf{v}_n$ is the n th parameter filter acting on the correlated channel of $\mathbf{X}$, and $*$ is the convolution operator. Since the output $\mathbf{U}$ is generated by summarizing the results of all channels, the relationship between these channels is implicitly and locally captured by the filters. In order to increase the sensitivity of the neural network to informative features used in the transformation, the learning of convolutional features

is enhanced by SE blocks to access global information and recalibrate parameters of the filter.

In the SE block, a global average pooling layer is employed to squeeze global information into one channel, which reduces the entire feature map to a single value by averaging all pixels in the feature map. The output $\mathbf{z} \in \mathbf{R}^C$ is produced by shrinking input U spatial dimensions, which can be calculated as:

$$\mathbf{z} = \mathbf{P}_{tr}(\mathbf{u}) = \frac{1}{H \times W} \sum_{i=1}^H \sum_{j=1}^W u_c(i, j). \quad (18)$$

After embedding the global information in the local channel descriptor through the squeeze operation, another operation is applied to capture channel-wise dependencies. This operation must satisfy two criteria: (i) learning nonlinear interactions between channels; and (ii) multiple channels can be activated. To limit the complexity and generalization of the model, a simple gating mechanism is added in the SE block using two fully connected (FC) layers and two activation functions, namely the sigmoid and rectified linear unit (ReLU) activation functions. Thus, the SE block can be interpreted as:

$$\mathbf{s} = \sigma(\mathbf{W}_2 \delta(\mathbf{W}_1 \mathbf{z})), \quad (19)$$

where $\mathbf{W}_1$ and $\mathbf{W}_2$ are weight matrices of two FC layers. Also, δ and σ are the ReLU and sigmoid activation functions, respectively.

In standard SE-Net architectures such as VGGNet [61], the SE block is inserted after each convolution layer. Here, we integrate the SE block with a residual neural network (ResNet) architecture [62], where the non-identity branch of the residual block is assumed to be the SE block transformation. Figure 8 schematically shows the SE-ResNet block used in the current study. In this model, the regular residual module is realized by a forward neural network with shortcut connections to perform identity mapping and directly transport outputs of the previous layer to the new layer. By applying this identity mapping, networks with a large number of layers can be easily trained, i.e., without introducing a significant training error and at a reasonable computational cost. A SE block composed of one global averaging layer, two FC layers, and two activation layers is added to the residual network. In this block, the global average pooling layer squeezes the global information with dimension $H \times W \times C$ into one channel ($1 \times 1 \times C$), then two hidden layers reduce the output by a reduction factor r leading to C/r neurons. The output is projected back to the same dimensional space as the input. Finally, the output of the SE-ResNet block is the sum of its inputs after applying average pooling to the output of the SE block.

More details regarding the SE-ResNet architecture used to predict the failure response of corroded pipe SVEs are illus-

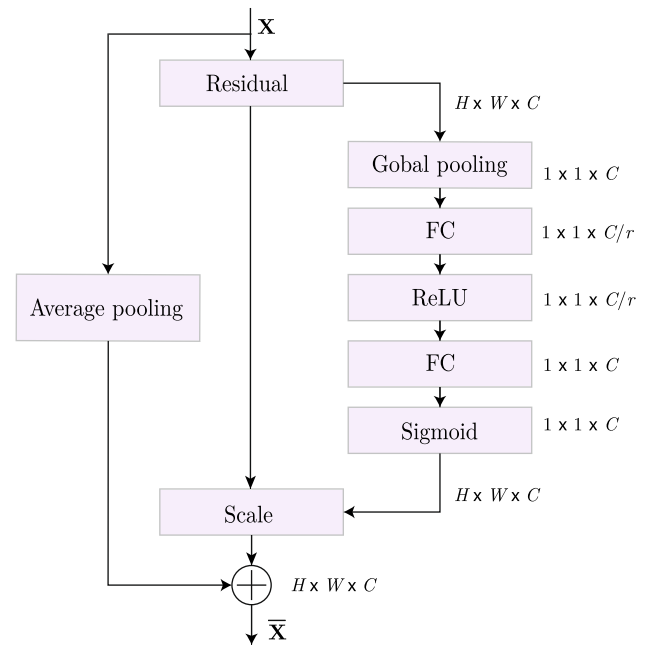


Fig. 8 Schematic representation of a SE-ResNet block

trated in Fig. 9. The model transforms $200 \times 200 \times 1$ grayscale images of the corroded SVE surface to a lower-dimensional vector of $1 \times 1 \times 2$ corresponding to the strength and toughness of the SVE in each channel. As shown in Fig. 9, a 2D convolutional layer is added to the model, followed by 6 SE-ResNet blocks, one dropout layer, one global average pooling layer, one activation layer, and 3 dense layers. The architecture of the first SE-ResNet block (number of filters/layers, output image size, etc.) is given in Table 3. Note that each convolutional block is composed of one convolutional layer followed by one batch normal layer and one ReLU activation layer in this SE-ResNet architecture. It is worth mentioning that other SE-ResNet blocks have similar architectures as the first block. Also, the rectified Adam is used as the optimizer for updating the model parameters. Compared to traditional optimizers such as Adam and SGD, the rectified Adam can achieve high accuracy using fewer epochs. The SE-ResNet model used here was developed in Python 3.8 and implemented in Keras [63], which is a high-level DL toolkit serving as a wrapper for TensorFlow [64].

4.3 Training and optimizing hyperparameters

The 40,000 data points generated after augmentations (SVE grayscale image labeled with strength/toughness values) were randomly split into 3 sets, where 30,000 samples were used for training the SE-ResNet model, 5000 samples for validation, and the remaining 5000 samples as the test set. For both training and testing data, the model's performance was measured using the mean squared error (MSE), the nor-

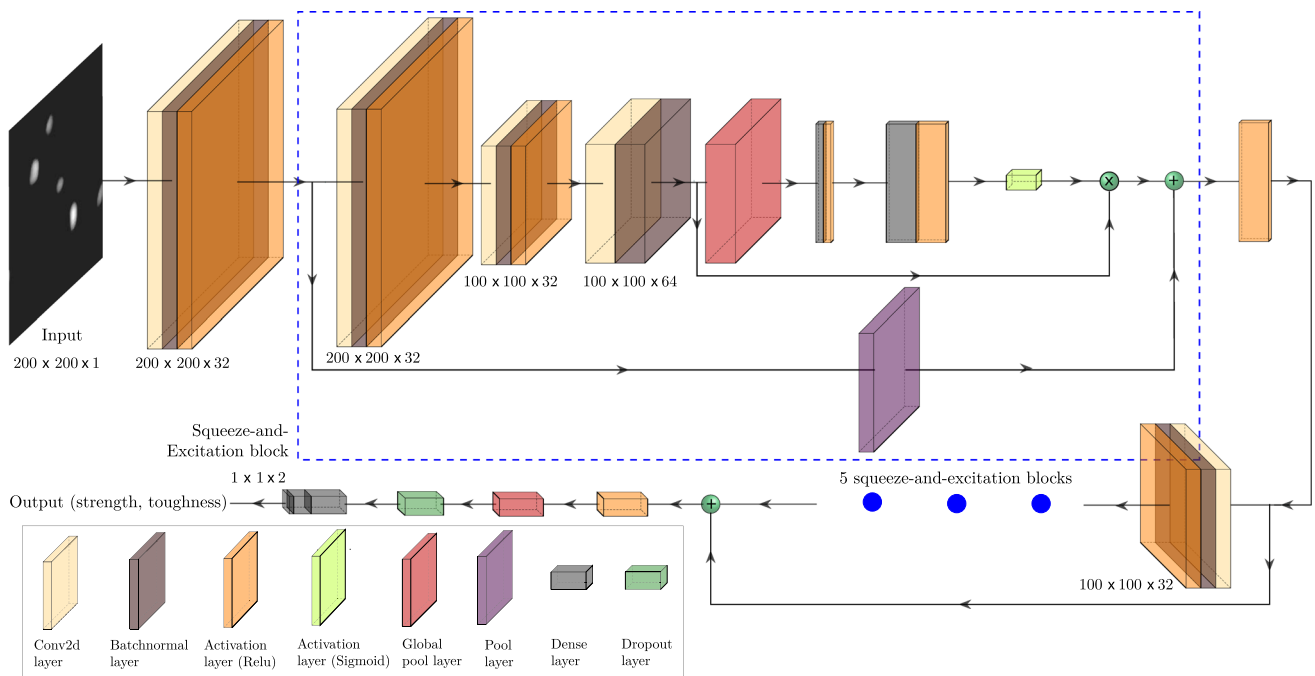


Fig. 9 Overall architecture of the SE-ResNet network used in the present study

Table 3 Architecture of the first SE-ResNet block

Layer name	Number of filters	Filter size	Output size
Input image	–	–	$200 \times 200 \times 1$
Convolutional block 1	32	7×7	$200 \times 200 \times 32$
Convolutional block 2	32	3×3	$100 \times 100 \times 32$
Convolutional block 3	64	3×3	$100 \times 100 \times 64$
Global pool layer	–	2×2	–
Dropout 1	–	0.2	–
Dense layer 1	16	–	16
Dense layer 2	64	–	64
Dropout 1	–	0.2	–
Reshape	64	–	$1 \times 1 \times 64$
Multiply	–	–	$100 \times 100 \times 64$
Average pool layer	–	2×2	–
Add	–	–	$100 \times 100 \times 64$

malized mean absolute error (NMAE), and the normalized maximum error (NMAX). To reduce the maximum error, MSE is chosen as the loss function in this SE-ResNet model. The MSE of the entire model (E_w^s) is the sum of MSE's associated with the strength (E_s^s) and the toughness (E_t^s) of each SVE, which is calculated as [64].

$$E_w^s = E_s^s + E_t^s = \frac{1}{N} \sum_{i=1}^N \left(|S_i - \hat{S}_i|^2 + |E_i - \hat{E}_i|^2 \right), \quad (20)$$

where N is the number of samples in the selected set, and S_i and $\hat{S}_i$ are true strength and the SE-ResNet prediction of

strength for the i th sample, respectively. Also, E_i and $\hat{E}_i$ denote the true toughness and predicted toughness of the i th sample, respectively. NMAE indicates the average error between projected values and ground truth values (evaluated via FE simulations) for a given batch of data. The NMAE of the entire model (E_w^a) can be written as

$$E_w^a = E_s^a + E_t^a = \frac{1}{N} \sum_{i=1}^N \left(\left| \frac{S_i - \hat{S}_i}{S_i} \right| + \left| \frac{E_i - \hat{E}_i}{E_i} \right| \right), \quad (21)$$

where E_s^a and E_t^a are the MAE of the failure strength and the toughness, respectively.

Table 4 The hyperparameter optimization (HPO) of the SE-ResNet model

Hyperparameter	Initial range	Optimal value (after HPO)
Number of SE blocks	[3, 8]	6
Number of convolutional blocks in one SE block	[2, 5]	3
Number of dense layers in one SE block	[1, 4]	2
Dropout rate	[0.0, 0.4]	0.24
Training data batch size	16, 32, 64	32
Batch normalization decay	[0.9, 0.99]	0.9

Table 5 k -fold cross-validation of the SE-ResNet model for predicting SVEs strength and toughness values

Model id	NMAE: training data		NMAX: training data		NMAE: test data		NMAX: test data	
	Strength (%)	Toughness (%)	Strength (%)	Toughness (%)	Strength (%)	Toughness (%)	Strength (%)	Toughness (%)
1	0.078	0.139	0.271	0.482	0.114	0.203	0.392	0.701
2	0.083	0.148	0.315	0.554	0.123	0.212	0.371	0.663
3	0.084	0.150	0.331	0.561	0.132	0.236	0.363	0.649
4	0.091	0.163	0.343	0.573	0.125	0.223	0.404	0.722
5	0.071	0.127	0.242	0.414	0.112	0.201	0.358	0.641

Determining the optimal architecture and hyperparameters of the SE-ResNet model used in the current study is critical to its predictive capability. Therefore, we examined different depths and numbers of filters in each convolution layer of this model to investigate their impact on the accuracy and the cost of training. Due to the large number of hyperparameters used in this model, the Bayesian optimization approach (BO) was implemented to determine the optimal combination of hyperparameters, as reported in Fig. 9. BO is a sequential method for optimizing parameters of any black-box function $f(\mathbf{x})$ [65] by creating a surrogate probability model $\hat{f}(\mathbf{x})$ that maps hyperparameters to a likelihood of a score on the objective function $P(y|\mathbf{x})$ based on prior prediction values (prior belief). Next, new combinations of hyperparameters $\mathbf{x}_n$ are generated and the combination that has the best performance on the surrogate model is selected by $\hat{f}(\mathbf{x})$. The selected hyperparameters are applied to the true objective function to evaluate the accuracy of this black-box function, after which the surrogate model is updated with new results. These steps are recursively repeated until the stopping criterion is reached.

For DL applications, the three most widely used techniques for building the surrogate probability function are the Gaussian Process [46], Random Forest regressor [44], and Tree-structured Parzen Estimator [43]. The Gaussian Process method generates surrogate probability functions by specifying random prior distributions following Gaussian distribution. Random Forest regressor builds the surrogate function by combining multiple regression trees and randomly generating feature sampling from the dataset for each tree. Instead of defining a predictive distribution, Tree-structured Parzen Estimator (TPE) is a sequential model-based optimization

(SMBO) approach. Based on prior measurements, SMBO builds a model to approximate the performance of hyperparameters, and then picks new hyperparameters to test on this model. In this work, TPE is utilized to build the surrogate probability function due to its faster convergence rate compared to the other two methods. The performance of a DL model is represented by the black-box function $f(\mathbf{x})$, which is usually highly non-convex. Although the performance of $f(\mathbf{x})$ can be measured at any point $\mathbf{x}$, evaluating $f(\mathbf{x})$ requires running the entire training cycle and therefore could be computationally expensive. In this work, the HPO process required more than 80 training cycles. Considering the computational cost associated with tuning hyperparameters, we limited the number of hyperparameters to six and set certain ranges for them. The initial ranges and optimized values of these hyperparameters are given in Table 4.

4.4 SE-ResNet Performance

After optimizing the SE-ResNet architecture using HPO, a k -fold cross-validation test was conducted to determine the accuracy of the model on new data not seen during the training process. As the first step, the entire data set (40,000 samples) was randomly shuffled and divided into k groups. One group was then taken as the test data, while the other $k - 1$ groups were considered as training/validation data sets. The model with optimized hyperparameters was trained with the new training data set. This process was repeated until all $k - 1$ groups were used as test data. In this work, k is set as 5, meaning the training model is trained and tested on 5 different data sets. Each data set is composed of 28,000 samples

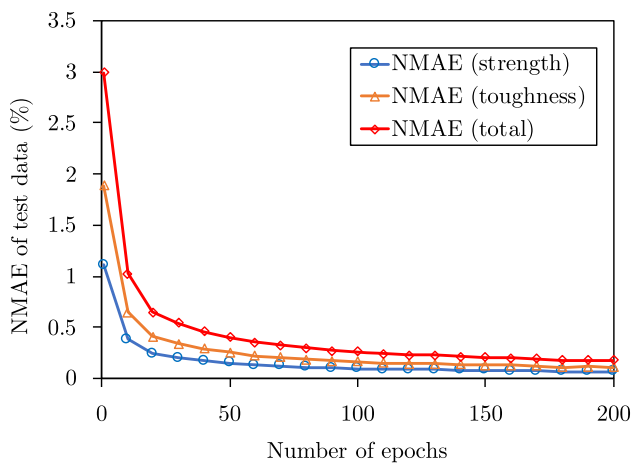


Fig. 10 The testing curve of the Se-ResNet model

as the training set, 4000 samples as the validation set, and 8000 samples as the test set.

The result of this cross-validation study is presented in Table 5, showing NMAE values of strength ranging from 0.112% to 0.132% and its NMAX values varying between 0.35 and 0.40% in the test data. The NMAE of toughness varies between 0.201 and 0.236%, while the toughness NMAE falls in the range of 0.641% to 0.722% for the test data. This study clearly shows an excellent performance of the SE-ResNet model for predicting the strength and toughness of corroded pipe SVEs. To have a better perception of the SE-ResNet performance, Fig. 11 compares predicted values of strength and toughness using Model 1 versus FE simulation results for 100 randomly selected SVEs from the test set. The testing curve of the SE-ResNet model is illustrated in Fig. 10, indicating the total NMAE value reduces from 3.0 to 0.4% after 50 epochs, while the reduction rate becomes much lower after 100 epochs and becomes close to 0 after 200 epochs.

In addition to reducing the labor cost associated with the FE modeling process (mesh generation), the key advantage of the SE-ResNet model is the ability to reduce the time associated with predicting SVEs strength and toughness by several orders of magnitude without the loss of accuracy. A comparison between these run times and corresponding accuracies for one SVE is provided in Table 6, where one P100 NVIDIA GPU with 16GB memory is used for SE-ResNet predictions versus an Intel Xeon 8268s with 24 cores for parallel FE simulations. Here, the SE-ResNet model performance is compared with two different FE models, one with a fine mesh composed of > 1 million elements and the second with a coarse mesh composed of 0.1 million elements. Note that the simulation results from FE models with a similar refinement level as the fine mesh are used for training the SE-ResNet model. As expected, Table 6 shows the significantly lower computational cost of predicting SVEs strength/toughness using SE-ResNet (0.07 s) compared to

the FE model with a fine mesh (1551 s) with a negligible error. Also, although the simulation time (353 s) of the FE model relying on the coarse mesh is still significantly higher than that of SE-ResNet, the former leads to a notable drop in accuracy. It is worth mentioning that the run times reported for FE simulations does not include the time spent on the mesh generation phase.

Figure 11 provides a comparison between strength and toughness of SVEs with corrosion pits obtained from FE simulations vs. predictions made by the SE-ResNet model. Note that the x -axis denotes the SVE number ranging from 1 to 50, while the y -axis shows the predicted strength or toughness value. The performance of the DL model for the training and test data are illustrated in Fig. 11a, b, respectively. Note that the normalized maximum error for the strength prediction on the training data is less than 0.3%, whereas it is less than 0.4% on the test data. Similarly, the normalized maximum error for predicting the toughness on the training data is less than 0.5%, whereas it is less than 0.7% on the test data.

4.5 Comparison with other DL models

The SE-ResNet model used for predicting the failure response of corroded pipe SVEs was selected after carefully examining the performance of several other DL algorithms in this work. Here, it is worthwhile to compare the performance of 5 DL models trained to predict the strength and stiffness of these SVEs, namely VGG-16, Inception, ResNet, DenseNet, and SE-ResNet. VGG-16 is a CNN architecture composed of convolution layers of 3x3 filters with a stride 1 and maxpool layers of 2x2 filters with a stride 2 [61]. The VGG16 model tested in this work has 16 layers and 45 million parameters. Instead of stacking convolution layers very deep, the Inception network introduces one inception module composed of multiple sizes of filters followed by a max-pool layer and concatenate layer [66]. We used the Inception V3 network with 4 branches in one Inception block to predict strength and stiffness values. The ResNet model uses heavy batch normalization and skip connections to transform the information from initial layers to deeper layers [62]. This forward transformation alleviates the vanishing-gradient problem, which is especially helpful when the CNN model has a deep architecture. In this work, we studied the performance of the ResNet-50 architecture, which has 50 layers and 32 million parameters. Densely Connected Convolutional Networks (DenseNet) concatenates the output feature maps of the layer with the next layer rather than using their summation. The DenseNet model tested in this work has three 5-layer dense blocks and 14 million parameters. It is worth noting that hyper-parameters of all DL models tested here were optimized using the BO method.

Table 7 provides a comparison between the accuracy (NMAE and NMAX), complexity (number of parameters),

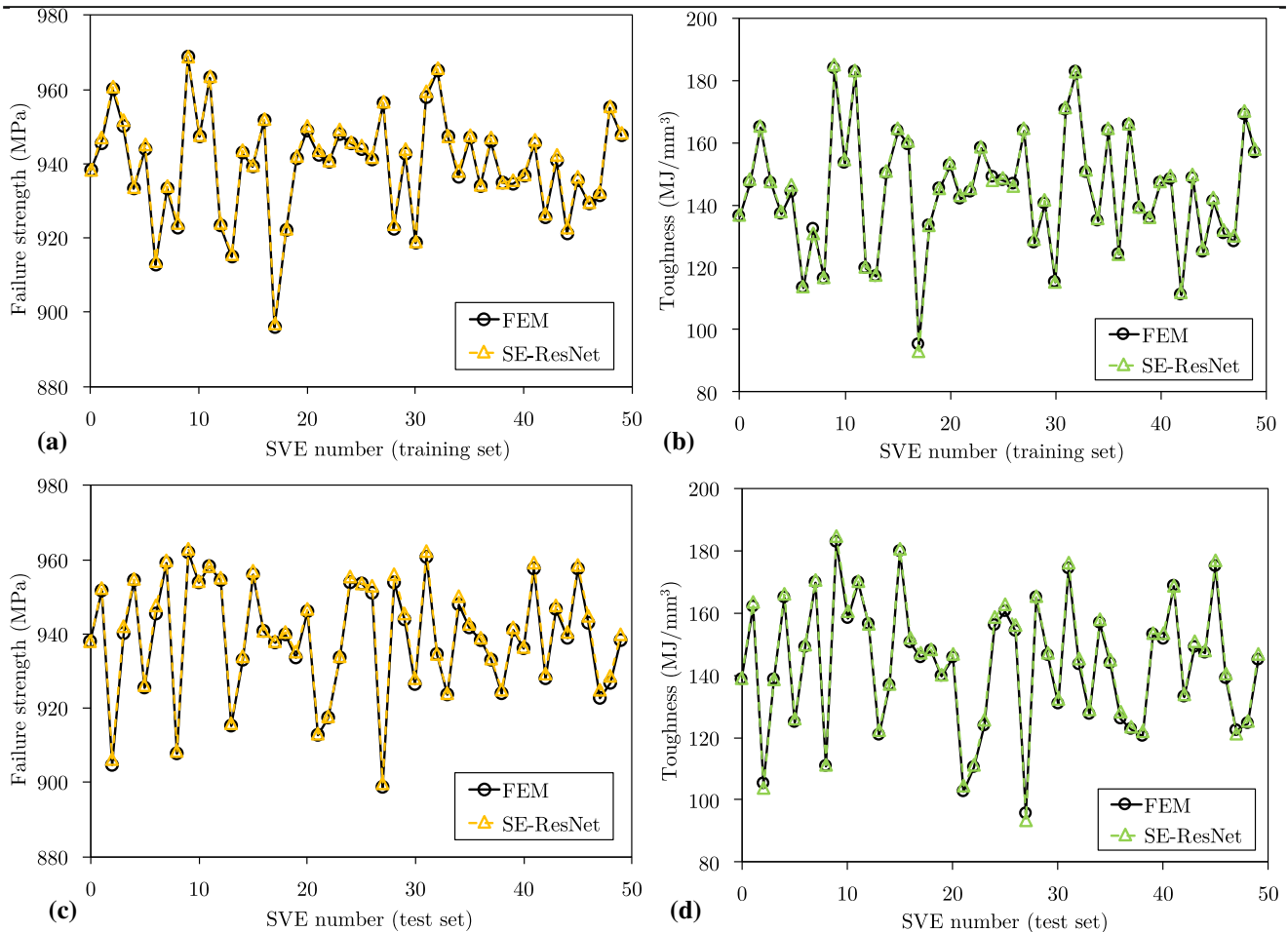


Fig. 11 Comparison between FEM and SE-ResNet predictions of strength and toughness of 50 randomly selected samples from **a, b** training data and **c, d** test data

Table 6 Comparison between run times of FE simulations (fine and coarse meshes) and SE-ResNet model for predicting the strength and toughness of SVEs

	FE: fine mesh	FE: coarse mesh	SE-ResNet
Failure strength (MPa)	915.2	926.5	917.1
Toughness (MJ/mm ³)	141.4	143.2	142.0
Time (s)	1,551	353	0.07

and the computational cost associated with training each DL model. Note that the NMAEs associated with VGG-16, Inception V3, ResNet-50, and DenseNet are 310%, 266%, 36.6%, and 96.4% higher than that of the SE-ResNet model. More importantly, a similar comparison with respect to NMAX values shows the errors are 358%, 293%, 39.1%, and 111% higher, respectively. Also, while the SE-ResNet model yields the best accuracy, it has the least number of training parameters, resulting in the lowest training time, as reported in Table 7.

Note that the CNN-based architectures studied in this work are tested only for one type of material (X100 steel) with corrosion pits. To predict the failure response of corroded SVEs made of different materials, the deep learning model

must be trained with new data corresponding to different material properties. The model itself must also be modified to receive material properties as part of input parameters via a new branch after the convolutional/pooling layers applied to the SVE image.

5 Conclusion

A deep learning model relying on the SE-ResNet architecture was presented for predicting the strength and toughness of steel pipes under the pitting corrosion attack directly from grayscale images of the corroded metal surface. An automatic computational framework, relying on

Table 7 Comparison between the performance of the SE-ResNet model and other DL frameworks for predicting the strength and toughness of corroded pipe SVEs

Model	Number of parameters	Training time for 200 epochs (s)	NMAE: test data		NMAX: test data	
			Strength (%)	Toughness (%)	Strength (%)	Toughness (%)
VGG-16	4.5×10^7	76,000	0.46	0.82	1.64	2.92
Inception	3.6×10^7	61,200	0.41	0.73	1.41	2.51
ResNet-50	3.2×10^7	52,800	0.15	0.27	0.50	0.89
DenseNet	1.4×10^7	51,000	0.22	0.39	0.76	1.34
SE-ResNet	9.2×10^6	43,000	0.11	0.20	0.36	0.64

virtual microstructure reconstruction played a pivotal role in acquiring the training data for the model by synthesizing and simulating the failure response of 10,000 3D SVEs. After data augmentation, 40,000 labeled data (corroded surfaces labeled with FE approximation of strength/toughness) were used for training/validating the SE-ResNet model. We implemented a Bayesian optimization approach to determine the model hyperparameters and optimized its microstructure, followed by a k -fold cross-validation study to ensure the model yields an acceptable performance. We showed that the SE-ResNet model can accurately predict corroded SVEs failure response ($< 1\%$ error) while providing orders of magnitude speedup compared to high-fidelity FE simulations. We also compared the performance of SE-ResNet model to 4 other widely used CNN-based models, i.e., VGG-16, Inception V3, ResNet-50 and DenseNet, showing the superior accuracy and lower training cost of this model for this problem.

Acknowledgements This work has been supported by the Air Force Office of Scientific Research (AFOSR) under grant number FA9550-21-1-0245. The authors also acknowledge the allocation of computing resources by the Ohio State University Simulation Innovation and Modeling Center (SIMCenter) and the Ohio Supercomputer Center (OSC).

References

1. Rajabipour A, Melchers RE (2013) A numerical study of damage caused by combined pitting corrosion and axial stress in steel pipes. *Corros Sci* 76:292–301
2. Frankel GS (1998) Pitting corrosion of metals: a review of the critical factors. *J Electrochem Soc* 145(6):2186
3. Davidson R (2002) An introduction to pipeline pigging, vol 9. Pigging Products and Services Association
4. Gupta A, Sircar A (2016) Introduction to pigging & a case study on pigging of an onshore crude oil trunkline. *J Latest Technol Eng Manag Appl Sci* 5(2):18–25
5. Papenfuss S (2009) Pigging the “unpiggable”: new technology enables inline inspection and analysis for non-traditional pipelines. In: 5 th MENDT conference, Bahrain
6. Nagaraj J (2013) Smart pigging in high pressure gas pipeline practical problems and solutions: a case study. In: ASME India oil and gas pipeline conference, vol 45349. American Society of Mechanical Engineers, p V001T02A005
7. Lucas EJK, Hales A, McBryde D, Yun X, Quarini GL (2017) Non-invasive ultrasonic monitoring of ice pigging in pipes containing liquid food materials. *J Food Process Eng* 40(1):e12306
8. Paik JK, Lee JM, Ko MJ (2003) Ultimate compressive strength of plate elements with pit corrosion wastage. *Proc Inst Mech Eng Part M J Eng Marit Environ* 217(4):185–200
9. Paik JK, Lee JM, Ko MJ (2004) Ultimate shear strength of plate elements with pit corrosion wastage. *Thin Walled Struct* 42(8):1161–1176
10. Zhang Y, Huang Y, Liu G (2008) A study on assessment of ultimate strength of ship structural plate with pitting corrosion damage. In: The eighth ISOPE Pacific/Asia offshore mechanics symposium, OnePetro

11. Ahmmad M, Sumi Y et al (2010) Strength and deformability of corroded steel plates under quasi-static tensile load. *J Mar Sci Technol* 15(1):1–15
12. Miller AG (1988) Review of limit loads of structures containing defects. *Int J Press Vessels Pip* 32(1–4):197–327
13. Harlow DG, Wei RP (1999) Probabilities of occurrence and detection of damage in airframe materials. *Fatigue Fract Eng Mater Struct* 22(5):427–436
14. Chen GS, Wan K-C, Gao M, Wei RP, Flournoy TH (1996) Transition from pitting to fatigue crack growth-modeling of corrosion fatigue crack nucleation in a 2024-t3 aluminum alloy. *Mater Sci Eng A* 219(1–2):126–132
15. Turnbull A, McCartney LN, Zhou S (2008) A model to predict the evolution of pitting corrosion and the pit-to-crack transition incorporating statistically distributed input parameters. In: Shipilov S, Jones R, Olive J-M, Rebak R (eds) *Environment-induced cracking of materials*. Elsevier, Amsterdam, pp 19–45
16. Amiri M, Arcari A, Airolidi L, Naderi M, Iyyer N (2015) A continuum damage mechanics model for pit-to-crack transition in aa2024-t3. *Corros Sci* 98:678–687
17. Xiao Y-C, Li S, Gao Z (1998) A continuum damage mechanics model for high cycle fatigue. *Int J Fatigue* 20(7):503–508
18. Hu P, Meng Q, Hu W, Shen F, Zhan Z, Sun L (2016) A continuum damage mechanics approach coupled with an improved pit evolution model for the corrosion fatigue of aluminum alloy. *Corros Sci* 113:78–90
19. Wang Y, Zheng Y (2019) Research on the damage evolution process of steel wire with pre-corroded defects in cable-stayed bridges. *Appl Sci* 9(15):3113
20. Hu WP, Shen QA, Zhang M, Meng QC, Zhang X (2012) Corrosion-fatigue life prediction for 2024-t62 aluminum alloy using damage mechanics-based approach. *Int J Damage Mech* 21(8):1245–1266
21. Ohga M, Appuhamy JMRS, Kaita T, Fujii K, Dissanayake PBR (2010) Numerical study on remaining strength prediction of corroded steel bridge plates. In: *International conference on sustainable built environments (ICSBE-2010)*, pp 529–536
22. Cui J, Yang F, Yang T-H, Yang G-F (2019) Numerical study of stainless steel pitting process based on the lattice Boltzmann method. *Int J Electrochem Sci* 14:1529–1545
23. Bruère VM, Bouchonneau N, Motta RS, Afonso S, Willmersdorf RB, Lyra PR, Torres JV, de Andrade EQ, Cunha DJ (2019) Failure pressure prediction of corroded pipes under combined internal pressure and axial compressive force. *J Braz Soc Mech Sci Eng* 41(4):1–10
24. Fu B, Stephens D, Ritchie D, Jones CL (2001) Methods for assessing corroded pipeline—review, validation and recommendations. Report GRTC, 3281
25. Choi JB, Goo BK, Kim JC, Kim YJ, Kim WS (2003) Development of limit load solutions for corroded gas pipelines. *Int J Press Vessels Pip* 80(2):121–128
26. Mohd MH, Lee BJ, Cui Y, Paik JK (2015) Residual strength of corroded subsea pipelines subject to combined internal pressure and bending moment. *Ships Offshore Struct* 10(5):554–564
27. Bhaduri A, He Y, Shields MD, Graham-Brady L, Kirby RM (2018) Stochastic collocation approach with adaptive mesh refinement for parametric uncertainty analysis. *J Comput Phys* 371:732–750
28. Bhaduri A, Brandyberry D, Shields MD, Geubelle P, Graham-Brady L (2020) On the usefulness of gradient information in surrogate modeling: application to uncertainty propagation in composite material models. *Probab Eng Mech* 60:103024
29. Bhaduri A, Meyer CS, Gillespie JW Jr, Haque BZ, Shields MD, Graham-Brady L (2021) Probabilistic modeling of discrete structural response with application to composite plate penetration models. *J Eng Mech* 147(11):04021087
30. Ok D, Pu Y, Incecik A (2007) Computation of ultimate strength of locally corroded unstiffened plates under uniaxial compression. *Mar Struct* 20(1–2):100–114
31. Ok D, Pu Y, Incecik A (2007) Artificial neural networks and their application to assessment of ultimate strength of plates with pitting corrosion. *Ocean Eng* 34(17–18):2222–2230
32. Yang Z, Yabansu YC, Al-Bahrani R, Liao WK, Choudhary AN, Kalidindi SR, Agrawal A (2018) Deep learning approaches for mining structure-property linkages in high contrast composites from simulation datasets. *Comput Mater Sci* 151:278–287
33. Rao C, Liu Y (2020) Three-dimensional convolutional neural network (3d-cnn) for heterogeneous material homogenization. *Comput Mater Sci* 184:109850
34. Mozaffar M, Bostanabad R, Chen W, Ehmann K, Cao J, Bessa MA (2019) Deep learning predicts path-dependent plasticity. *Proc Natl Acad Sci* 116(52):26414–26420
35. Haghighat E, Raissi M, Moure A, Gomez H, Juanes R (2020) A deep learning framework for solution and discovery in solid mechanics. [arXiv:2003.02751](https://arxiv.org/abs/2003.02751)
36. Elshaw R, Maher M, Sakr S (2019) Automated machine learning: state-of-the-art and open challenges. [arXiv:1906.02287](https://arxiv.org/abs/1906.02287),
37. Kuhn M, Johnson K et al (2013) *Applied predictive modeling*, vol 26. Springer, Berlin
38. Yang L, Shami A (2020) On hyperparameter optimization of machine learning algorithms: theory and practice. *Neurocomputing* 415:295–316
39. Talbi E-G (2021) Automated design of deep neural networks: a survey and unified taxonomy. *ACM Comput Surv: CSUR* 54(2):1–37
40. Bardenet R, Brendel M, Kégl B, Sebag M (2013) Collaborative hyperparameter tuning. In: *International conference on machine learning*. PMLR, pp 199–207
41. Li L, Jamieson K, Rostamizadeh A, Gonina E, Ben-Tzur J, Hardt M, Recht B, Talwalkar A (2020) A system for massively parallel hyperparameter tuning. *Proc Mach Learn Syst* 2:230–246
42. Joy TT, Rana S, Gupta S, Venkatesh S (2016). Hyperparameter tuning for big data using Bayesian optimisation. In: *2016 23rd international conference on pattern recognition (ICPR)*. IEEE, pp 2574–2579
43. Bergstra J, Bardenet R, Bengio Y, Kégl B (2011) Algorithms for hyper-parameter optimization. In: *Proceedings of the 24th international conference on neural information processing systems*. pp 2546–2554
44. Hutter F, Hoos HH, Leyton-Brown K (2011) Sequential model-based optimization for general algorithm configuration. In: *International conference on learning and intelligent optimization*. Springer, pp 507–523
45. Bergstra J, Bengio Y (2012) Random search for hyper-parameter optimization. *J Mach Learn Res* 13(2):281–305
46. Snoek J, Larochelle H, Adams RP (2012) Practical bayesian optimization of machine learning algorithms. In: *Proceedings of the 25th international conference on neural information processing systems*, vol 2. pp, 2951–2959
47. Hill R (1985) On the micro-to-macro transition in constitutive analyses of elastoplastic response at finite strain. In: *Mathematical proceedings of the Cambridge Philosophical Society*, vol 98. Cambridge Univ Press, pp 579–590
48. Lemaitre J (1985) A continuous damage mechanics model for ductile fracture. *J Eng Mater Technol* 107:83–89
49. Bataille J, Kestin J (1979) Irreversible processes and physical interpretation of rational thermodynamics. *J Non Equilib Thermodyn* 4:229–258
50. Hooputra H, Gese H, Dell H, Werner H (2004) A comprehensive failure model for crashworthiness simulation of aluminium extrusions. *Int J Crashworthiness* 9(5):449–464

51. Tanguy B, Luu TT, Perrin G, Pineau A, Besson J (2008) Plastic and damage behaviour of a high strength X100 pipeline steel: Experiments and modelling. *Int J Press Vessels Pip* 85(5):322–335
52. Keshavarz A, Ghajar R, Mirone G (2014) A new experimental failure model based on triaxiality factor and Lode angle for X-100 pipeline steel. *Int J Mech Sci* 80:175–182
53. Dzuga J, Spaniel M, Prantl A, Konopik P, Ruzicka J, Kuzelka J (2018) Identification of ductile damage parameters for pressure vessel steel. *Nucl Eng Des* 328:372–380
54. Ahmadian H, Yang M, Nagarajan A, Soghrati S (2019) Effects of shape and misalignment of fibers on the failure response of carbon fiber reinforced polymers. *Comput Mech* 63(5):999–1017
55. Yang M, Garrard J, Abedi R, Soghrati S (2021) Effect of microstructural variations on the failure response of a nano-enhanced polymer: a homogenization-based statistical analysis. *Comput Mech* 67:315–340
56. Yang M, Nagarajan A, Liang B, Soghrati S (2018) New algorithms for virtual reconstruction of heterogeneous microstructures. *Comput Methods Appl Mech Eng* 338:275–298
57. Soghrati S, Nagarajan A, Liang B (2017) Conforming to interface structured adaptive mesh refinement: new technique for the automated modeling of materials with complex microstructures. *Finite Elem Anal Des* 125:24–40
58. Nagarajan A, Soghrati S (2018) Conforming to interface structured adaptive mesh refinement: 3d algorithm and implementation. *Comput Mech* 62(5):1213–1238
59. Brain D, Webb GI (1999) On the effect of data set size on bias and variance in classification learning. In: *Proceedings of the fourth Australian Knowledge Acquisition Workshop*, University of New South Wales, pp 117–128
60. Hu J, Shen L, Sun G (2018) Squeeze-and-excitation networks. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp 7132–7141
61. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. [arXiv:1409.1556](https://arxiv.org/abs/1409.1556)
62. He K, Zhang X, Ren S, Sun J (2016). Deep residual learning for image recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp 770–778
63. Chollet F et al (2015) Keras
64. Abadi M, Barham P, Chen J, Chen Z, Davis A, Dean J, Devin M, Ghemawat S, Irving G, Isard M et al (2016) Tensorflow: a system for large-scale machine learning. In: *12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16)*, pp 265–283
65. Shahriari B, Swersky K, Wang Z, Adams RP, De Freitas N (2015) Taking the human out of the loop: a review of bayesian optimization. *Proc IEEE* 104(1):148–175
66. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2015) Going deeper with convolutions. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp 1–9

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.