

**ARTICLE TYPE**

# Deep convolutional neural networks for uncertainty propagation in random fields

Xihai Luo<sup>1</sup> | Ahsan Kareem<sup>1</sup><sup>1</sup>NatHaz Modeling Laboratory, University of Notre Dame, Notre Dame, IN, USA**\*Correspondence**X. Luo Email: xluo1@nd.edu;  
A. Kareem Email: kareem@nd.edu**Summary**

The development of a reliable and robust surrogate model is often constrained by the dimensionality of the problem. For a system with high-dimensional inputs/outputs (I/O), conventional approaches usually use a low-dimensional manifold to describe the high-dimensional system, where the I/O data is first reduced to more manageable dimensions and then the condensed representation is used for surrogate modeling. In this study, a new solution scheme for this type of problems based on a deep learning approach is presented. The proposed surrogate is based on a particular network architecture, i.e. convolutional neural networks. The surrogate architecture is designed in a hierarchical style containing three different levels of model structures, advancing the efficiency and effectiveness of the model in the aspect of training. To assess the model performance, uncertainty quantification is carried out in a continuum mechanics benchmark problem. Numerical results suggest the proposed model is capable of directly inferring a wide variety of I/O mapping relationships. Uncertainty analysis results obtained via the proposed surrogate have successfully characterized the statistical properties of the output fields compared to the Monte Carlo estimates.

**KEYWORDS:**

Random field, High dimensionality, Surrogate model, Deep learning, Uncertainty quantification

## 1 | INTRODUCTION

The primitive parameters of real-world mechanical systems such as material properties and external excitations are often not known exactly and are subject to uncertainty. This uncertainty may arise from inevitable manufacturing imperfections, incomplete knowledge on system parametrization, and others Beer, Ferson, & Kreinovich (2013); Dai & Cao (2017); Dai, Zhang, & Wang (2015); Sadeghi, Fayek, & Pedrycz (2010). In most cases, uncertain parameters are modeled as random variables utilizing a sequence of probability distributions and the system of interest is characterized by stochastic partial differential equations (SPDEs). Because the input parameters are random, the output of the system also becomes a random variable. To obtain comprehensive probabilistic descriptions

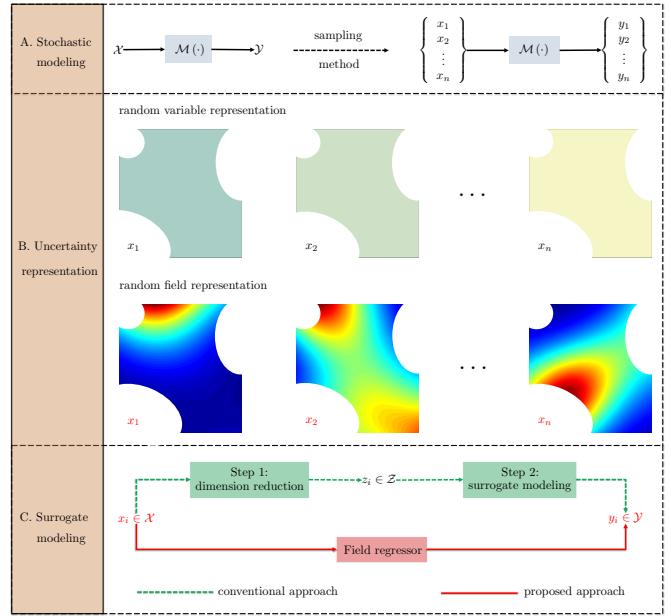
for the random output quantities, one can rely on the ensemble of independent samples drawn from the distribution of the inputs using the Monte Carlo method, evaluating the system at a finite number of realizations and estimating the statistics of the target outputs (See Fig. 1 .(A)). Such uncertainty quantification (UQ) task involving many system queries can be greatly expedited through the use of surrogate models especially for applications that depend on a time-demanding simulator Forrester, Sobester, & Keane (2008); Viana, Simpson, Balabanov, & Toropov (2014).

Conceptually simple, a formidable challenge remains open concerning the development of a reliable and robust surrogate in cases where the input/output is no longer a random variable, instead, is a random field or even a set of random fields. For instance, one can treat material properties as random variables, which are random in essence but are uniformly distributed over the domain. Such treatment ignores the possible spatial

variation and the realization of perfectly uniform material is unlikely to occur in practice, thus fails to provide a realistic reflection of the real-world situations Babuška, Nobile, & Tempone (2007); G. Chen, Zhang, Rasmussen, & Fan (2016); Gurley & Kareem (1999). On the other hand, the random field theory offers a more reliable avenue as it systematically takes the spatial correlation into accounts (See Fig. 1 .(B)). The numerical discretization of a random field with small correlation can easily result in a large number of random variables. Unfortunately, most existing surrogate models dealing with uncertainty propagation tasks have difficulty scaling to high-dimensional problems and the conjunction use of the random field and the time-demanding computer model can be prohibitive Babuška et al. (2007); Dai & Cao (2017); Schwab & Gittelson (2011).

Given that the direct construction of surrogate models yielding an easy-to-evaluate mechanism between high-dimensional inputs and outputs can be extremely demanding, practitioners usually attempt to project the given random field data from its original high dimension to a lower dimension. The standard approach to performing dimensionality reduction on the continuous functional space in the UQ literature is the Karhunen-Loève expansion (KLE) Ghanem & Spanos (2003); Le Maître & Knio (2010); Xiu & Karniadakis (2002). Known under different names such as principal component analysis, proper orthogonal decomposition, etc, the KLE first projects a random field, which is infinite dimensional, to finite dimensions using eigenfunctions of the random field's covariance function as basis functions. Then, the KLE represents the random field as a linear combination of these orthogonal eigenfunctions and their corresponding uncorrelated coefficients, which are random variables. In spite of numerous applications in which the KLE based dimensionality reduction scheme circumvents the curse of dimensionality, the KLE by definition can only preserve the first and second order statistics, i.e., the mean and covariance of the random field. However, uncertainties arise from material heterogeneities or randomly distributed force fields usually exhibit nonlinear structures Kareem (2008). Moreover, the KLE is an unsupervised learning algorithm from a machine learning perspective Murphy (2012), indicating it does not explicitly integrate the output information into the dimensionality reduction procedure and additional classification/regression model is needed to complete the surrogate modeling task (See the upper green route in Fig. 1 .(C)).

In recent years, deep neural networks (DNNs) have gained increased popularity in representing the high-dimensional solutions of a wide class of partial differential equations T. Q. Chen, Rubanova, Bettencourt, & Duvenaud (2018); Mo, Zhu, Zabaras, Shi, & Wu (2019); Raissi, Perdikaris, & Karniadakis (2019); Sirignano & Spiliopoulos (2018); Zhu &



**FIGURE 1** A. Stochastic modeling using the Monte Carlo method; B. Random field theory to model the spatial variability; C. Surrogate modeling for random field uncertainty propagation

Zabaras (2018). Compared to the conventional surrogate modeling approaches, DNNs tackle the curse of dimensionality by learning the latent representation through the use of a series of layers, which are connected by affine operations and nonlinear activations. Usually formed as a supervised learning problem, DNNs are capable of providing an end-to-end field-to-field surrogate model, thus avoiding the need to build a dimension reduction model (See the lower red route in Fig. 1 .(C)). In this work, a deep encoder-decoder network is proposed to empower the UQ analysis of civil structures with spatially varying system properties. Convolutional neural networks (CNNs) with innovative connection architectures and advanced resizing techniques are utilized for the direct learning of the intrinsic high-dimensional mapping. Within the realm of neural networks modeling, CNNs are the workhorse of image classification Krizhevsky, Sutskever, & Hinton (2012); Rafiei & Adeli (2017), object detection Cha, Choi, & Büyüköztürk (2017); Gopalakrishnan, Khaitan, Choudhary, & Agrawal (2017), computer vision Yang et al. (2018); Yeum & Dyke (2015), and signal processing Nabian & Meidani (2018); Rafiei & Adeli (2018). This type of neural networks gains computational efficiency in dealing with data of a grid-like topology by extracting multi-scale features within the modeling process. This unique learning property triggers the use of CNNs for surrogate modeling of the complex nonlinear mapping between discretized random fields.

In this paper, a machine learning based structural analysis framework is proposed to statistically characterize the system responses under various spatially varying properties. Compared to conventional CNNs based models, the proposed surrogate model is designed in a hierarchical form, greatly improving the training efficiency. Specifically, additional network connections between nonadjacent convolutional layers are created in such a way that extracted features can be reused, thus reducing the model parameters and strengthening the machine learning process He, Zhang, Ren, & Sun (2016a 2016b); G. Huang, Liu, Van Der Maaten, & Weinberger (2017). Moreover, the operations of deconvolution included in the CNNs can generate checkerboard artifacts, especially in the context of regression problems of high-resolution images. To address this issue, the upsampling method bicubic resizing is integrated into the convolution operation in the decoding process Dong, Loy, He, & Tang (2015); Odena, Dumoulin, & Olah (2016). To make the training process more efficient, the annealing learning strategy is incorporated into the setting of stochastic gradient descent and batch normalization method is adopted to avoid data distortion Ioffe & Szegedy (2015); Robbins & Monro (1951). The overall machine learning methodology is applied to a plate-like structure governed by a set of fourth-order elliptic partial differential equations. The flexibility and robustness of the proposed surrogate model is demonstrated by considering one-to-one ( $\mathbb{R}^{4096} \rightarrow \mathbb{R}^{4096}$ ), one-to-many ( $\mathbb{R}^{4096} \rightarrow \mathbb{R}^{12288}$ ), and many-to-many ( $\mathbb{R}^{8192} \rightarrow \mathbb{R}^{8192}$ ) mappings, where each one represents a discretized high-dimensional random field.

In the rest of this paper, Section 2 gives the problem statement for using deep neural networks to propagate uncertainties represented by random fields. Section 3 provides a comprehensive guide in terms of building, training, and validating the proposed surrogate. In Section 4, a systematic case study on first identifying intrinsic relationships between discretized random fields and then applying the learned surrogate to the uncertainty analysis is presented. Lastly, Section 5 summarizes the principal conclusions and discusses the future work.

## 2 | PROBLEM STATEMENT

### 2.1 | Problem formulation: boundary value problem

This paper focuses on the development of surrogate models for stochastic and high-dimensional systems Babuška et al. (2007); Schwab & Gittelson (2011); Xiu & Karniadakis (2002). To fulfill this objective, general boundary value problems of the following elliptic PDEs form are considered:

$$\mathcal{L}_s(\Theta(s), u(s)) = f(s) \quad \text{in } S \quad (1)$$

where  $s$  is the cartesian coordinates,  $\mathcal{L}_s$  is a differential operator,  $\Theta$  represents an input property field encompassed in the constitutive equation,  $u$  denotes an unknown solution to the equation,  $f$  represents the source term that drives the system, and  $S$  is a Lipschitz smooth domain. To determine a unique solution, boundary conditions, i.e. the Dirichlet and Neumann boundary conditions are imposed on  $\Gamma_D \subset \partial S$  and  $\Gamma_N \subset \partial S$ , respectively:

$$\begin{aligned} u &= g^{(D)} & \text{at } \partial\Gamma_D \\ \nabla u \cdot \hat{n} &= g^{(N)} & \text{at } \partial\Gamma_N \end{aligned} \quad (2)$$

with  $\hat{n}$  denoting the unit normal on the boundary. The operator  $\nabla$  is considered only with respect to (w.r.t.) the spatial variables  $s$ . In a typical probabilistic mechanics problem, one often deals with systems subjected to stochastic excitations  $f(s)$ , or themselves involve random parameters  $\Theta(s)$ , or both.

*Remark 1.* 1. In the context of uncertainty quantification of structural systems, the operator  $\mathcal{L}_s$  could be the momentum balance equation, which reads the mass density, body force, initial velocity, etc as the inputs and computes the corresponding displacement field.

### 2.2 | Proposed approach: image-to-image regressor

The goal is to build an accurate but fast-to-evaluate surrogate model to accelerate the propagation and quantification of the influence of spatially varying uncertainties  $f(s)$  and  $\Theta(s)$  on structural behaviors  $u$ . First, random fields are used to account for the stochasticity in Eq. (1) arises from the spatially varying  $f(s)$  and  $\Theta(s)$  fields Shinozuka & Jan (1972). Let  $(\Omega, \mathcal{S}, \mathbb{P})$  be a complete probability space, where  $\Omega$  is the set of the elementary events  $\omega$ ,  $\mathcal{S}$  is an  $\sigma$ -algebra on the set  $\Omega$ , and  $\mathbb{P}$  is a probability measure. A random field  $r(s, \omega)$  is a manifold in the Hilbert space consisting of a family of random variables. Hence, the solution mapping of Eq. (1) in a geometrically bounded regime with discretized dimension  $d$  is expressed as:

$$u : S \times \Omega \rightarrow \mathbb{R}^d \quad (3)$$

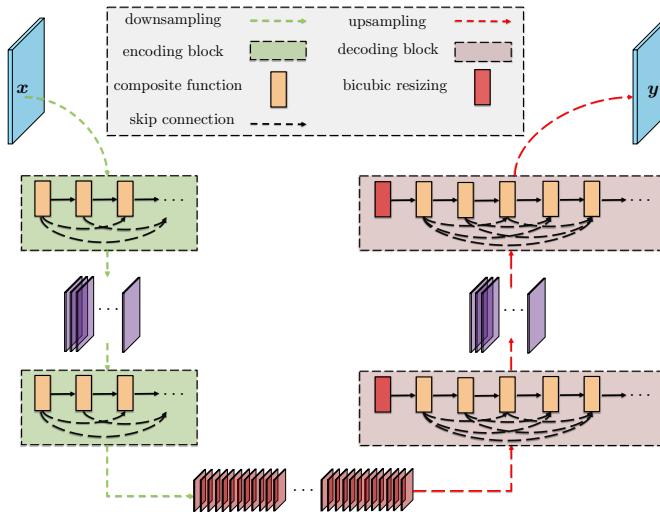
The next step is to build a surrogate model in aid of the uncertainty propagation of the spatial variability. To achieve this goal, deep neural networks (DNN) are considered:

$$y = f(x) \xrightarrow{\approx} \hat{y} = \hat{f}_{DNN}(x) \quad (4)$$

where the input  $x = [\Theta, f]$  and  $y$  refers to the solution field  $u$ .

To address the high-dimensional mapping between the discretized stochastic input fields and the corresponding output fields, the surrogate model is designed in an image-to-image transformation form. In distinction from an image classification problem that only requires a single label regarding the

entire image, the image regression problem entails the pixel-wise predictions. Consequently, the model should be able to faithfully persevere local features regarding spatial variability in addition to the global features in terms of the skeleton of the data. An advanced network architecture is hereby proposed to address this changeling surrogate modeling problem, where state-of-the-art techniques such as the composite layer, bicubic resizing, and skip connections have been seamlessly integrated into the deep convolutional neural networks. A graphic illustration of this image-to-image regressor is shown in Fig. 2 and the details on the architecture design and hyperparameters identification are provided in the next section.



**FIGURE 2** Schematic diagram of the proposed approach

### 3 | METHODOLOGY

This section covers a guideline on the components, architecture, and training of the proposed surrogate. Because emphasis is placed on the regression problems of random fields, the model is hereby named as *field regressor*.

#### 3.1 | Model ingredients

**Composite function.** Following state-of-the-art network designs such as AlexNet Krizhevsky et al. (2012), VGG Simonyan & Zisserman (2014), U-Net Ronneberger, Fischer, & Brox (2015), ResNet He et al. (2016a), etc, the essential model component is defined as a composite function of Batch Normalization (BN) Ioffe & Szegedy (2015), rectified linear units (ReLU) Glorot, Bordes, & Bengio (2011), and

Convolution (Conv) LeCun, Bottou, Bengio, Haffner, et al. (1998):

$$\Upsilon(\mathbf{x}) \doteq (\text{Conv} \circ \text{ReLU} \circ \text{BN})(\mathbf{x}) \quad (5)$$

where  $\circ$  is taken to be the composition operator. Compared to conventional *Conv* or empirical mixed layer designs, composite function enabled deep architectures tends to provide compelling accuracy He et al. (2016b).

**Skip connection.** According to Eq. (5), neural networks approximate the input-output relationship  $f(\cdot)$  as:

$$\hat{f}(\mathbf{x}) = \Upsilon^L \circ \Upsilon^{L-1} \circ \dots \circ \Upsilon^1(\mathbf{x}) \quad (6)$$

where  $L$  is the number of layers defined in the network architecture. Usually, CNNs take the output of the  $(l-1)^{\text{th}}$  convolutional layer as the input to the  $(l)^{\text{th}}$  layer:

$$\mathbf{x}_l = \mathbf{y}_{l-1} = \Upsilon_{l-1}(\mathbf{x}_{l-1}) \quad (7)$$

Though simple and neat, recent research reveals such block design often encounters information degradation problems, which impede the training of CNNs Glorot & Bengio (2010). To remedy this issue, distant connections among nonadjacent layers are established, where every single layer is fully connected with all the subsequent layers He et al. (2016a); G. Huang et al. (2017). Consequently, the improved expression of function writes as:

$$\mathbf{x}_l = \mathbf{y}_{l-1} = \Upsilon_{l-1}([\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_{l-1}]) \quad (8)$$

Because each  $\Upsilon_i (i = 1, 2, \dots, l-1)$  extracts  $k$  feature maps, the input field for the  $l^{\text{th}}$  layer has  $C_l = k_0 + k(l-1)$  input features. Noticeably,  $k_0$  denotes the number of random fields of the first layer and  $k$  is the growth rate.

**Resizing upsampling.** Projecting condensed low-resolution feature maps to the high-dimensional output space of interest in the context of pixel-wise regression is among the main challenges in computer graphics. Image resizing techniques are adopted as conventional deconvolution confronts the checkerboard pattern of artifacts due to its uneven overlap property Odena et al. (2016). Specifically, methods such as bilinear interpolation, nearest-neighbor interpolation, bicubic interpolation, and area interpolation have been tested and bicubic resizing that considers 16 pixels ( $4 \times 4$  translated pixel per axis) has been selected for its superior performance.

#### 3.2 | Model architecture

**Architecture design.** Motivated by the network designs in recent works such as fully convolutional neural networks (FCNNs) Long, Shelhamer, & Darrell (2015) and SegNet Badrinarayanan, Kendall, & Cipolla (2017), the proposed field regressor has a downsampling process and a corresponding

upsampling process, where each sampling process contains a set of encoding/decoding blocks. The overall architecture is illustrated in Fig. 2 . Specifically, each encoding block in the downsampling process performs the composition operation introduced in Eq. (5) to produce a set of feature maps. To reduce the number of trainable model parameters, feature maps produced by the composition function are continually down-sized to a coarser spatial resolution. This is achieved by adopting non-unit stride in the convolution operation Dumoulin & Visin (2016). Compared to other techniques such as max pooling, the use of non-unit stride can better preserve sub-regional data relations, which are of crucial importance for the pixel-wise regression. Meanwhile, the decoding block upsamples its input feature maps using the bicubic resizing method, followed by a composite layer. Skip connections are only considered between nonadjacent composite layers coming from the same block. Hence, each block produces dense feature maps via stacking the convolved outputs provided by the composition functions G. Huang et al. (2017).

**Hyperparameter identification.** The main hyperparameters regarding deep learning based pixel-wise regression cover can be broadly classified into two categories: (1) *configuration hyperparameters* include number of layers, kernel size, stride number, zero padding, growth rate, and optimizer selection; (2) *learning hyperparameters* include learning rate, dropout ratio, weight decay, and batch size. First, grid searching is adopted for determining optimized hyperparameters in terms of model configuration. For instance, the spatial dimension of receptive fields at the coarsest level. It was found a resolution ratio of 1/16 has the best performance in general across different datasets. In particular, each encoding block scales the size of its input data down by 2 through specifying the stride number of the convolution operation included in the first composite function to 2. Similarly, the decoding block upsamples the extracted feature maps by 2 using bicubic resizing at the beginning of the block. Secondly, random searching is adopted for identifying training hyperparameters such as initial learning rate, weight decay, and etc Bergstra & Bengio (2012). Specifically, a bandit-based approach that is so-called Hyperband is implemented to speed up the standard random search alright via adaptive resource allocation and early-stopping Li, Jamieson, DeSalvo, Rostamizadeh, & Talwalkar (2016).

### 3.3 | Model training

**Regularized loss function.** Training a field regressor centers on optimizing the network parameters  $\theta$  in such a way that the discrepancy between model predictions  $\hat{y}_i$  and true observations  $y_i$  is minimized. For regression tasks, the discrepancy is typically measured by the mean squared error (MSE) loss.

Given a training dataset  $\mathcal{D} = \{\mathbf{x}_i, \mathbf{y}_i\}_{i=1}^N$  consisting of  $n$  independent and identically distributed (i.i.d.) samples, the loss function is defined as:

$$\mathcal{L}(\theta; \mathcal{D}) = \frac{1}{N} \sum_{i=1}^N \|\mathbf{y}_i - \hat{f}(\mathbf{x}_i, \theta)\|^2 \quad (9)$$

In practice, neural nets are notorious for overfitting. One usually resorts to regularization techniques, penalizing  $f'(\cdot)$  by either promoting sparsity or driving weights to 0. In this paper,  $L_2$  norm is chosen to penalize the model complexity Goodfellow, Bengio, & Courville (2016); Y. Huang, Beck, & Li (2018). The regularized MSE loss writes as:

$$\mathcal{L}(\theta; \mathcal{D}) = \frac{1}{N} \sum_{i=1}^N \|\mathbf{y}_i - \hat{f}(\mathbf{x}_i, \theta)\|^2 + \lambda \sum_{j=1}^K \|\mathbf{W}^{(j)}\|_2^2 \quad (10)$$

where  $K$  is the total number of layers, and  $\lambda$  is a small constant that forces  $f'(\cdot)$  learning small weights (often termed as the weight decay), and  $\mathbf{W}^j$  denotes all the parameters in the  $j^{th}$  layer. For the field regressor,  $\mathbf{W}^j$  includes the kernel weights utilized in the convolution operation, the scale and shift parameters in the batch normalization operation, and bicubic interpolation parameters used in the image resampling.

**Mini-batch stochastic gradient descent.** In order to compute the parameters  $\theta$ , the gradient-based optimization is adopted as  $f'(\cdot)$  is a highly composite function involving a series of layers of nonlinear transformation Rumelhart, Hinton, Williams, et al. (1988). In particular, the Mini-batch stochastic gradient descent (mini-batch SGD) is implemented to solve this problem Goodfellow et al. (2016). The central idea is to approximate the negative gradient of the loss function by means of the expectation of a batch of samples:

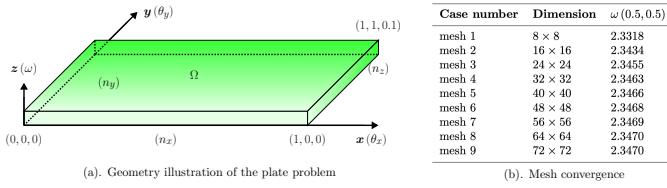
$$\theta_{k+1} \leftarrow \theta_k - \eta_k \nabla_{\theta} \mathcal{L}(\theta; \mathcal{D}_M) \quad (11)$$

where  $\eta_k$  is the learning rate in the  $k^{th}$  iteration and  $\mathcal{L}(\theta; \mathcal{D}_M)$  is the loss function evaluated using a mini-batch subset of samples  $\mathcal{D}_M \subset \mathcal{D}$ . Specifically, a mini-batch is initialized between 5 and 500 samples, chosen through a random manner and the Adaptive Moments (ADAM) optimization algorithm, a variant of the SGD methods, is selected to update the learnable parameters Kingma & Ba (2014):

$$\theta_{k+1} \leftarrow \theta_k + \eta_k \frac{M_k}{1 - \beta_1^k} / \left( \sqrt{\frac{V_k}{1 - \beta_2^k}} + \epsilon \right) \quad (12)$$

In Eq. (12),  $\epsilon$  is a small number introduced to prevent 0 denominator, and  $M_k$  and  $V_k$  are estimates of the mean and variance of the gradients, respectively. In ADAM, the update scheme of them are given as follows:

$$\begin{aligned} M_k &= \beta_1 M_{k-1} + (1 - \beta_1) \mathcal{L}(\theta; \mathcal{D}_M) \\ V_k &= \beta_2 V_{k-1} + (1 - \beta_2) \mathcal{L}^2(\theta; \mathcal{D}_M) \end{aligned} \quad (13)$$



(a). Geometry illustration of the plate problem

(b). Mesh convergence

**FIGURE 3** Illustration of the case study

Note that  $M_0$  and  $V_0$  are set to 0 during the initialization.  $\beta_1$  and  $\beta_2$  are averaging parameters. In this work, values are configured as  $\beta_1 = 0.9$  and  $\beta_2 = 0.999$  following the suggestion in Kingma & Ba (2014).

## 4 | CASE STUDY

### 4.1 | Problem setup

To demonstrate the effectiveness and efficiency of the proposed surrogate, the following benchmark Mindlin–Reissner model on the unit square domain Mindlin (1951); Reissner (1945) is considered:

$$\begin{aligned} -\operatorname{div} \mathbf{C}\boldsymbol{\varepsilon}(\boldsymbol{\theta}) - \gamma &= 0 & \text{in } \Omega \\ -\operatorname{div} \boldsymbol{\gamma} &= f & \text{in } \Omega \\ -\lambda t^{-2}(\nabla w - \boldsymbol{\theta}) &= \gamma & \text{in } \Omega \\ \boldsymbol{\theta} = 0, w = 0 & \quad \text{on } \partial\Omega \end{aligned} \quad (14)$$

where  $\Omega = [0, 1]^2 \subset \mathbb{R}^2$  is a smooth domain with a small thickness  $t = 0.1$ ,  $\mathbf{C}$  is the positive definite tensor denoting the bending moduli,  $\boldsymbol{\varepsilon}$  is the linear Green strain tensor,  $\boldsymbol{\theta} = [\theta_x, \theta_y]$  represents the rotations of the surface,  $w$  is the transverse displacement in z-direction,  $\gamma$  denotes the scaled shear stresses,  $f$  is the applied scaled transversal load, and  $\lambda = E\kappa/2(1 + \nu)$  is the shear modulus with  $E$  denoting the Young's modulus,  $\nu = 0.3$  denoting the Poisson ratio, and  $\kappa = 5/6$  denoting the shear correction factor. Essentially, Eq. (14) is a system of second order partial differential equations that can describe the bending behavior of a clamped plate loaded by a transverse force and Fig. 3 . (a) shows the computational domain of the interest.

The objective of this benchmark problem is to build a surrogate model that can propagate and quantify the spatially-varying uncertainty associated with model inputs. The selected random input property is modeled by means of random field theory where its logarithm is a Gaussian random field:

$$\log \mathbf{x}(s) \sim \text{GP}(\mathbf{m}(s), k(s, s')) \quad (15)$$

where  $\mathbf{m}(s)$  and  $k(s, s')$  are the mean and covariance functions of the Gaussian process. In this case study, the mean function is defined to be zero and the exponentiated quadratic

covariance function that is also known as the RBF kernel is adopted:

$$k(s, s') = \sigma^2 \exp\left(-\frac{(s - s')^2}{2\ell^2}\right) \quad (16)$$

where the correlation length  $\ell = 0.5$ . To demonstrate the flexibility and robustness of the proposed surrogate, the scope of the case study is expanded by applying the proposed approach to the modeling of different mapping relationships, even though there is no obvious structure sharing between the input and output fields.

### 4.2 | Training, testing, and validation datasets

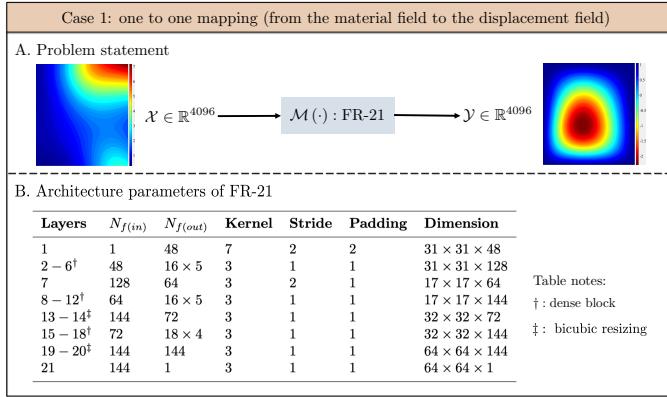
To collect data, the forward PDEs are solved using the finite element method (FEM) Hughes (2012). After the convergence study (See Fig. 3 . (b)), the unit square domain is discretized by  $64 \times 64$  quadrilateral isoparametric element (Q4) elements, ensuring the accuracy of the FE solution. Hence, the number of degrees of freedom of the high-fidelity FE model is 12675. The FEM writes the weak form of equilibrium equations (Eq. (14)) of the system into a set of algebraic equations, and solves the whole set by the Newton-Raphson method.

There are several sampling schemes available to generate data. In this case study, there are four sets of data: the training set, test set, validation set, and uncertainty analysis set. Specifically, each case contains training trials with 64, 128, 256, 512, and 1024 samples. The test and validation set each contains another 200 samples, and the reference statistics of the uncertainty analysis is computed by the Monte Carlo method using  $10^5$  samples.

In light of model training, metrics such as coefficient of determination ( $R^2$  value) and root mean squared error (RMSE) are adopted to evaluate the regression performance. In particular, they are evaluated for both the training and test data sets, and they are recorded every 10 epochs. For the propagated uncertainty, Monte Carlo method is implemented to approximate the mean, variance, and probability distributions of selected points. Predictive results via the trained surrogate are compared to the finite element results. Besides computing the difference between the ground truth and surrogate prediction, relative error defined as

$$\mathcal{E}(\mathbf{u}_{\text{FR}}, \mathbf{u}_{\text{FEM}}) = \frac{|\sum_{i=1}^{n_x} \sum_{j=1}^{n_y} |\mathbf{u}_{\text{FEM}}| - \sum_{i=1}^{n_x} \sum_{j=1}^{n_y} |\mathbf{u}_{\text{FR}}||}{\sum_{i=1}^{n_x} \sum_{j=1}^{n_y} |\mathbf{u}_{\text{FEM}}|} \quad (17)$$

is used to measure the prediction accuracy. The surrogate modeling algorithm is implemented in the open source machine learning library TensorFlow and the scripts are tested on a single NVIDIA GeForce GTX 1080 Ti X GPU. For the



**FIGURE 4** Neural networks architecture design and parameterization of FR-21

purpose of reproducibility, the code to replicate the optimization results in this work will be made available at <https://xihaiier.github.io> upon publication of this manuscript.

## 4.3 | Results

### 4.3.1 | Case 1: one-to-one mapping

The first case concerns the use of field regressor to learn one-to-one mapping. Let the input be the material field, i.e. the young modulus and the output be the displacement field. The model inputs  $x \in \mathbb{R}^{4096}$  are discretized random field realizations and the model outputs  $y \in \mathbb{R}^{4096}$  are the transverse deformation at the finite element centers. The uncertainty analysis of the target system with high-dimensional stochastic inputs covers two parts. First, a surrogate model  $\hat{f} : \mathbb{R}^{4096} \times [0, 1]^2 \rightarrow \mathbb{R}^{4096}$ , which can accurately map the snapshot of the material field to the displacement field governed by the PDEs is trained. Next, the trained surrogate is applied to the uncertainty quantification tasks, computing statistical moments and PDFs.

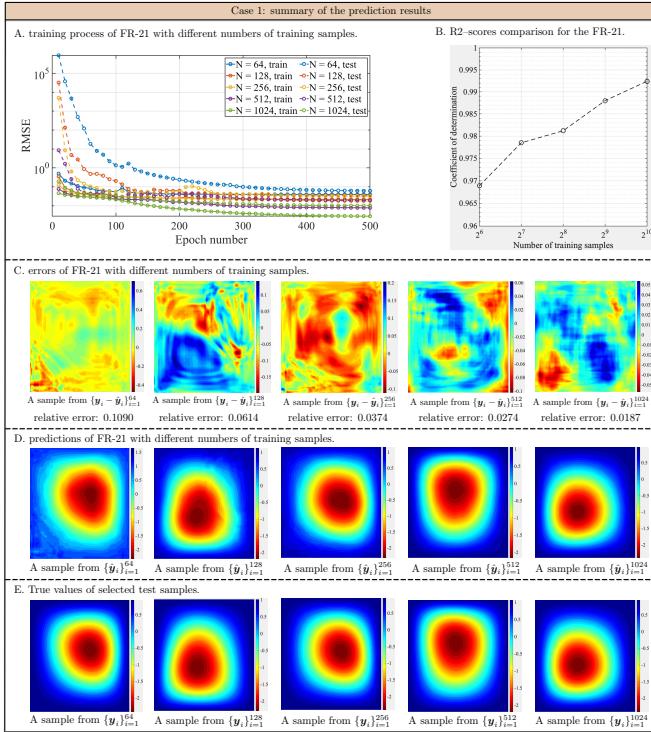
**Network architecture.** After extensive hyperparameter and network architecture search, the field regressor is built using 21 convolutional layers and this configuration is hence referred to as FR-21. Specifically, the dimension of each random field sample has been reduced twice by the downsampling processor  $64 \times 64 \rightarrow 31 \times 31 \rightarrow 17 \times 17$ . There are three dense blocks within FR-21. They share the same kernel size, stride number, and padding number while have different growth number and depth. Fig. 4 gives a general summary of the one-to-one mapping problem as well as the details of these network configuration parameters.

To learn the high-dimensional input-output relationship, the gradient-based optimizer Adam is selected. The surrogate model (FR-21) is trained with training data size varies from 64 to 1024. The initial learning rate is set to  $\eta_0 = 0.005$  and

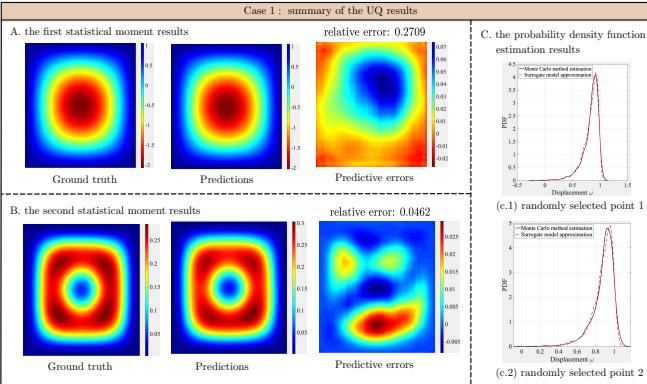
the step decay learning strategy (often known as annealing) is adopted to gradually lower the learning rate during training. The default value  $\eta_0$  is reduced by a annealing rate  $\zeta = 0.75$  every 20 epochs and the model trained with 500 epochs in total. For regularization, the weight decay is initialized to  $\lambda = 7 \times 10^{-6}$  and the batch size is 8.

**Optimization results.** Fig. 5 (A) plots the training process in terms of the RMSE value, which is recorded every 20 epochs. The rapidness and steadiness of the convergence of the FR-21 can be seen from the results. In particular, after around 300 epochs of the Adam optimizer, an approximately stabilized solution is achieved. The optimization algorithm converges faster with more training samples and the RMSE value of the testing dataset reduces as the training size increases:  $64 \rightarrow \dots \rightarrow 1024$ . Meanwhile, the coefficient of determination (also known as the  $R^2$ -score) is evaluated using 200 test samples with training dataset sizes. Note that the trained FR-21 shows a remarkable generalization ability where the  $R^2$ -score is more than 0.9 in all the trials. In Fig. 5 (B), the model achieves a relatively high  $R^2$ -score of 0.967 with only 64 training samples and the  $R^2$ -score gets close to 0.99 when increasing the training dataset to 512 samples, which indicates that the predicted output fields by FR-21 match the ground truth computed by the FEM very well. Fig. 5 (C) show some of the predicted results. The errors of the predicted solutions of FR-21 are compared with respect to the FE solutions. Note that the results representing the output fields have been standardized so that a more clear comparison can be carried out between FR-21 with different training data sizes. For visualization purpose, a sample is randomly selected from the testing dataset  $\mathcal{D}_{\text{test}} = \{\mathbf{x}_i, \mathbf{y}_i\}_{i=1}^{200}$ . Remarkably, the propsoed FR-21 is capable of providing good predictions. Even when using only 64 training samples, the model is able to quickly capture the landscape of the target output field. The prediction quality can be significantly improved as more samples are put into the training process.

**Uncertainty analysis results.** The uncertainty analysis results predicted by the FR-21 is verified against the solution computed by the FEM solver. The number of training samples is set to  $N = 1024$ , and the number of testing samples is set to  $M = 1 \times 10^5$ . Fig. 6 (A) and (B) show the the mean and variance of the displacement field, respectively. It can be seen that FR-21 achieves smooth contour estimation that is close to the Monte Carlo result. The maximum error of the estimate measured in a relative error form is less than 10%. Meanwhile, two points are randomly selected out of the domain and show the PDFs, respectively. The unbounded kernel smoothing function is used to estimate the PDFs, where the type of kernel smoother is set to normal. Fig. 6 (C) compares the PDFs at two locations from FR-21 and FEM solver. As desired, the PDFs obtained by the surrogate model at two randomly selected locations are almost identical to the reference solution.



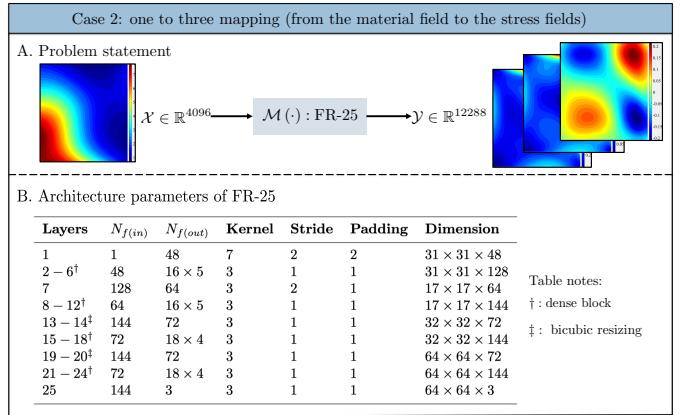
**FIGURE 5** Training and prediction performance of FR-21



**FIGURE 6** Uncertainty quantification results of FR-21

### 4.3.2 | Case 2: one-to-many mapping

In this case, the field regressor is generalized to the surrogate learning of one-to-many mapping. The input field is the young modulus  $\mathbf{x} = E(s)$ , and the output fields are stress fields that essentially are functions of displacement fields  $\mathbf{y} = [\sigma_v(s), \tau_{\max}(s), \tau_{xy}(s)]$ . In particular,  $\sigma_v$  is the Von Mises stress,  $\tau_{\max}$  denotes the maximum shear stress, and  $\tau_{xy}$  is the shear stress. They can be obtained by evaluating the quadrature points for the stress-strain relationship. The objective is to

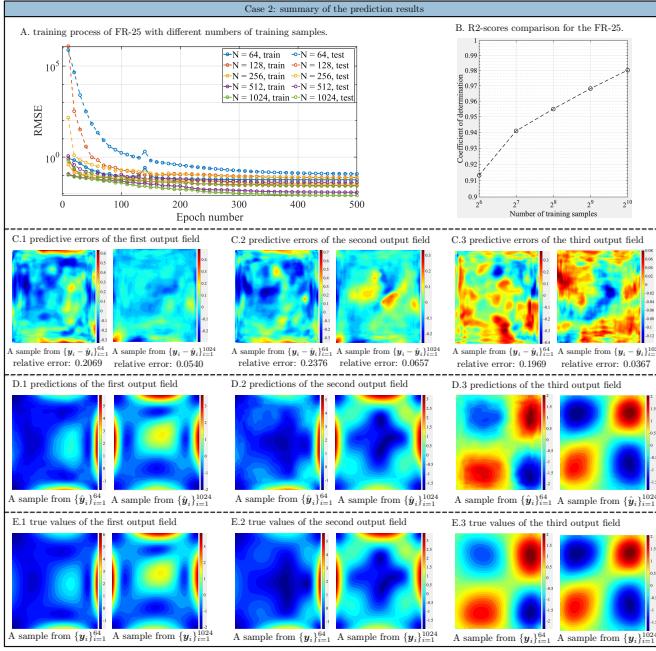


**FIGURE 7** Neural networks architecture design and parameterization of FR-25

build a surrogate model that can predict three discretized random fields simultaneously from one input random field  $\hat{f} : \mathbb{R}^{64 \times 64} \times [0, 1]^2 \rightarrow \mathbb{R}^{64 \times 64 \times 3}$ .

**Network architecture.** The surrogate model is derived using 25 convolutional layers in total, and hence is referred to as FR-25. There are four dense blocks in the FR-25. In particular, a dense block before the last convolutional layer is built, ensuring that sufficient information has been extracted to construct three independent output fields. The number of feature maps increases from 72 before the block to 144 after a set of block layers 21 – 24. Fig. 7 gives a graphic representation of the one-to-many mapping problem  $\mathbb{R}^{4096} \rightarrow \mathbb{R}^{12288}$  and summarizes the detailed designs of FR-25.

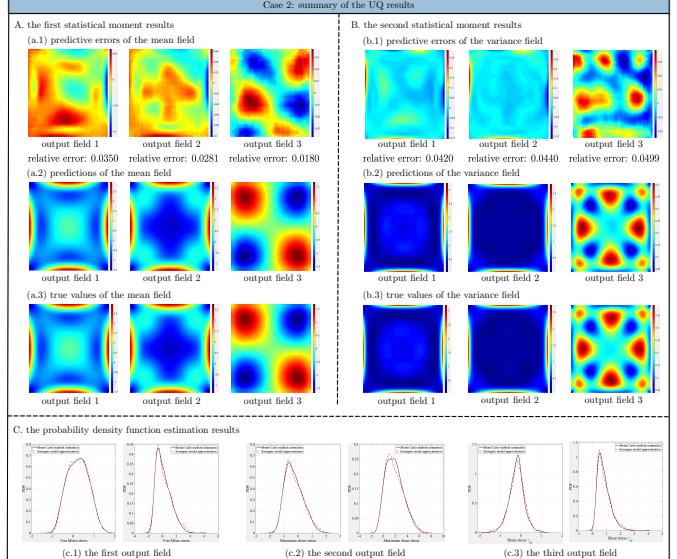
**Optimization results.** The surrogate model is trained with the same optimization configuration stated in Section 4.3.1. Fig. 8 (A) confirms the learning effectiveness of the proposed surrogate model. The Adam optimizer reaches stabilized state after 350 epochs. The objective function minimizes faster with more training data, and the RMSE value at epoch 500 increases when the training size decreases. Fig. 8 (B) shows the  $R^2$ -score for each training trial. Note that the  $R^2$ -score is evaluated using all the predicted fields, implying each sample  $\hat{\mathbf{y}}_i$  belongs to the real-valued vector space  $\mathbb{R}^{64 \times 64 \times 3}$ . The results show that the surrogate, FR-25, provides an accurate approximation of the high-dimensional input-output relationship even with a few samples (e.g. 64 samples). Specifically, these  $R^2$ -score values indicate the majority of the observed variation (approximately 90 percents) can be explained by the model. The predicted results using 64 and 1024 training samples are presented in Fig. 8 (C), respectively. The estimated stress fields via the FR-25 closely resemble the landscape of the ground truth data when only 64 samples are available for the model training. And the prediction accuracy improves substantially when the



**FIGURE 8** Training and prediction performance of FR-25

training sample size increases to 1024. Noticeably, the prediction results of the shear stress field  $\tau_{xy}(s)$  is relatively less smoother, especially when the training sample size is small. This is due to the weights assigned to each output field. Without any prior knowledge, the intuitive assignment is the use of uniform weights. The different performances regarding different output fields are largely alleviated as the size of training samples increases, and the predictions become very close to the reference solutions, where the relative maximum error of  $\sigma_v(s)$ ,  $\tau_{max}(s)$ , and  $\tau_{xy}(s)$  are in the same range.

**Uncertainty analysis results.** The predictive performance of the FR-25 to the machine learning with a small dataset seen in this case is a strong indication of the effectiveness and robustness of using the proposed surrogate modeling approach for uncertainty analysis. A number of  $1 \times 10^5$  independent and identically distributed samples have been generated for the computation of the reference statistics using the Monte Carlo method. With respect to the PDFs, 2 points are once again randomly selected from each output field and use kernel methods to estimate the corresponding PDF. For the mean field prediction, the maximum error is less than 5% measured in a relative error sense and the error centers around the extreme value region, for instance, the central cross region of the  $\tau_{max}(s)$  field (See Fig. 9 (A)). For the mean variance prediction, the maximum error is distributed on the boundary lines in terms of the prediction of  $\sigma_v(s)$  and  $\tau_{max}(s)$  and is overlapped with the extreme value region in the case of  $\tau_{xy}(s)$  (See Fig. 9 (B)). The PDFs obtained by the surrogate model at two randomly selected locations are almost identical to the reference solution.



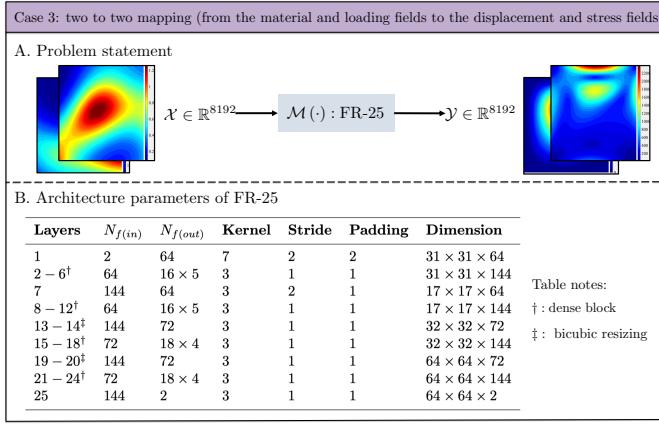
**FIGURE 9** Uncertainty quantification results of FR-25

Fig. 9 (C) shows the randomly selected points have non-gaussian distributions of the  $\sigma_v(s)$ ,  $\tau_{max}(s)$ , and  $\tau_{xy}(s)$  value. And the FR-25 can capture different PDFs with remarkably high precision.

### 4.3.3 | Case 3: many-to-many mapping

The third and final case study investigates the efficacy of the proposed surrogate modeling approach for many-to-many mapping problems. The boundary value problem defined by the elliptic PDEs (Eq. (14)) with multiple random input fields is considered. In particular, the material  $E(s)$  and loading  $f(s)$  properties are considered to vary spatially where the correlated variation has the same analytical form (Eq. (15)). Subsequently, interests are focused on characterizing the spatial variability induced effects on the structural performances that are denoted by the displacement  $w(s)$  and stress  $\sigma_v(s)$ . The input-output relationship satisfies the high-dimensional mapping  $\hat{f} : \mathbb{R}^{64 \times 64 \times 2} \times [0, 1]^2 \rightarrow \mathbb{R}^{64 \times 64 \times 2}$  on the unit square domain.

**Network architecture.** Similar to the previous case, the deep neural networks based surrogate model is configured with 25 convolutional layers. Since the model input  $x$  covers two independent random fields that are physically irrelevant, more features have been extracted and stacked by the end of the first layer, where the number of input and output features is 2 and 64, respectively. As displayed in the table of Fig. 10, a total of 4 dense blocks have been constructed on account for the sufficiency of the reconstruction of the output fields. The concept of the many-to-many mapping problem is illustrated in Fig. 10



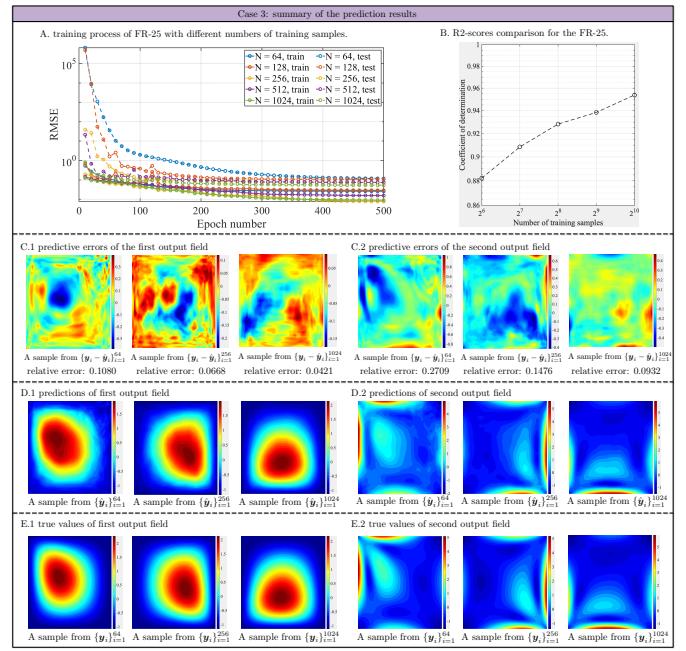
**FIGURE 10** Neural networks architecture design and parameterization of FR-25

(A) and the details of the FR-25 in the present case is given in Fig. 10 (B).

**Optimization results.** Fig. 11 (A) shows the optimization history of the model learning with different size of training samples. The strength of the proposed surrogate in terms of training is verified as the objective function decreases dramatically within the first 200 epochs. In comparing the different trials of this case study, again, it can be seen that an increase in the training sample size leads to a better-optimized result. Indeed, the  $R^2$ -score illustrates the importance of providing sufficient training samples, where the coefficient of determination value is relatively lower compared to the second case with the same amount of training samples, with the  $R^2$ -score of the trial using 64 samples slightly smaller than 0.9. This is due to the fact that the complexity of this case is higher than the second one. Potential improvement strategy includes altering the architecture design of the field regressor, for instant, extracting initial features in a separate manner rather than the joined style, that is,

$$\begin{aligned}
 \text{Joint extraction: } & 64 \times 64 \times 2 & \rightarrow \\
 & 31 \times 31 \times 64 \\
 \text{Separate extraction: } & 64 \times 64 \times 2 = \frac{64 \times 64 \times 1}{64 \times 64 \times 1} & \rightarrow \\
 & 31 \times 31 \times 32 = 31 \times 31 \times 64 & \\
 & & (18)
 \end{aligned}$$

For the purpose of easy comparison, a united design is used in all cases. The accuracy and consistency of the predicted results of the proposed approach can once again be seen from Fig. 11 (C), (D), and (E). In particular, the precision observed in the selected trials of this case study approximately equivalent to those seen in the previous cases, and thus indicates the robustness and strong potential of the proposed surrogate



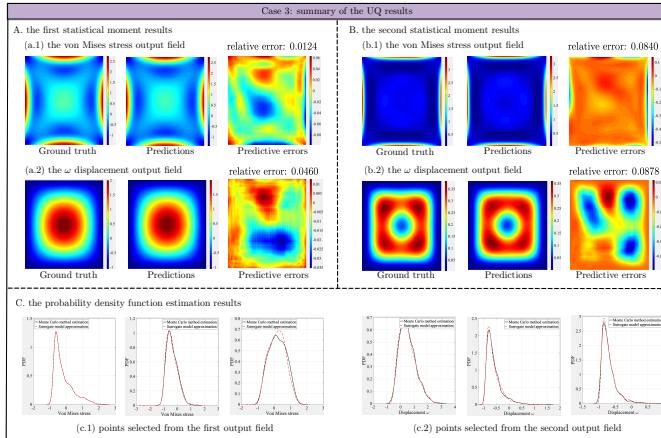
**FIGURE 11** Training and prediction performance of FR-25

modeling approach for solving a wide variety of problems of interest.

**Uncertainty analysis results.** Fig. 12 first investigates the prediction performance of the mean and variance fields by means of FR-25. The reference solution is obtained by numerically solving the governing SPDEs with the finite element scheme  $10^5$  times. Fig. 12 (A) and (B) present the good agreement between the surrogate modeling and high-fidelity FE results for the statistics of the target output fields. Fig. 12 (C) also depicts the PDFs of two output fields at three specific locations. It can be seen that the field regressor is able to accurately approximate the probabilistic response. This once again illustrates the robustness and efficiency of the proposed field regressor, i.e., using 1024 samples is sufficient to capture most of the stochasticity in this  $\mathbb{R}^{8192} \rightarrow \mathbb{R}^{8192}$  SPDEs.

## 5 | CONCLUDING REMARKS

This paper presents a deep neural network approach for uncertainty analysis of elliptic systems with spatially varying properties. Novelties of the proposed field regressor can be outlined in three perspectives. First, information flowing over the model is gradually downsized to a stack of feature maps. Model-estimated output fields are then reconstructed from extracted features. Such a multi-scaling data mechanism hereby advances the training process as fewer model parameters are required. Second, skip connections between nonadjacent convolutional layers are introduced to the basic



**FIGURE 12** Uncertainty quantification results of FR-25

network architecture, allowing previously extracted features to be reused in the subsequent layers. This property favors the machine learning process as a deeply structured architecture feasibly becomes trainable and numerically stable. Third, bicubic resizing is integrated into the conventional convolution operation in the upsampling process. Consequently, detailed variation patterns in the high-dimensional data can be better preserved.

Note the effectiveness of the proposed surrogate model is demonstrated on a plate-like structure with spatial variability. Admittedly, more work is needed to systematically evaluate the performance. In this regard, extensive work on the surrogate modeling of Poisson's equation, Darcy's law, and nonlinear geometric analysis via the proposed method is carried out and results are summarized as supplementary materials available at <https://xihaiier.github.io>. In a broader context, this work advocates a promising strategy for empowering the current handcraft feature engineering. For instance, pixel-wise crack detection, where spatial characteristics of cracks are vital in assessing the performance of existing structures. Current machine learning based crack detections mainly focus on binary prediction, neglecting pixel-wise impacts while the proposed model allows multi-scale segmentation and identification. Another application of practical interest centers on connecting the incoming turbulence to the aerodynamic pressure fields over tall buildings. Complex interactions take place as a turbulent flow field encounters a bluff structure like a building, resulting in pressure fluctuations on the building surface. The transformation between turbulence and pressure fields does not offer an analytical functional relationship which has led to experiments to estimate the pressure field from the turbulence input. The universal functional approximation ability of neural networks makes it a strong candidate for connecting these two fields.

Finally, it should be noted that the current surrogate is merely built through data without integrating any physics into the neural network architecture. Possible improvement strategies of the model performance may include utilizing well-established physical interpretations during the architecture design process of the surrogate. For instance, building a new neural network as a representation of governing equations or engineering constraints in addition to the current field regressor will be helpful.

## 6 | ACKNOWLEDGMENTS

X.L. acknowledges the support of NatHaz Modeling Laboratory at the University of Notre Dame where this work was performed. This research effort was supported in part by the National Science Foundation through Grant No. 1520817 and No. 1612843, and the funds provided by the Moran professorship.

## References

- Babuška, I., Nobile, F., & Tempone, R. (2007). A stochastic collocation method for elliptic partial differential equations with random input data. *SIAM Journal on Numerical Analysis*, 45(3), 1005–1034.
- Badrinarayanan, V., Kendall, A., & Cipolla, R. (2017). Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12), 2481–2495.
- Beer, M., Ferson, S., & Kreinovich, V. (2013). Imprecise probabilities in engineering analyses. *Mechanical systems and signal processing*, 37(1-2), 4–29.
- Bergstra, J., & Bengio, Y. (2012). Random search for hyperparameter optimization. *Journal of Machine Learning Research*, 13(Feb), 281–305.
- Cha, Y.-J., Choi, W., & Büyüköztürk, O. (2017). Deep learning-based crack damage detection using convolutional neural networks. *Computer-Aided Civil and Infrastructure Engineering*, 32(5), 361–378.
- Chen, G., Zhang, H., Rasmussen, K. J., & Fan, F. (2016). Modeling geometric imperfections for reticulated shell structures using random field theory. *Engineering structures*, 126, 481–489.
- Chen, T. Q., Rubanova, Y., Bettencourt, J., & Duvenaud, D. K. (2018). Neural ordinary differential equations. In *Advances in neural information processing systems* (pp. 6571–6583).
- Dai, H., & Cao, Z. (2017). A wavelet support vector machine-based neural network metamodel for structural reliability

- assessment. *Computer-Aided Civil and Infrastructure Engineering*, 32(4), 344–357.
- Dai, H., Zhang, H., & Wang, W. (2015). A multiwavelet neural network-based response surface method for structural reliability analysis. *Computer-Aided Civil and Infrastructure Engineering*, 30(2), 151–162.
- Dong, C., Loy, C. C., He, K., & Tang, X. (2015). Image super-resolution using deep convolutional networks. *IEEE transactions on pattern analysis and machine intelligence*, 38(2), 295–307.
- Dumoulin, V., & Visin, F. (2016). A guide to convolution arithmetic for deep learning. *arXiv preprint arXiv:1603.07285*.
- Forrester, A., Sobester, A., & Keane, A. (2008). *Engineering design via surrogate modelling: a practical guide*. John Wiley & Sons.
- Ghanem, R. G., & Spanos, P. D. (2003). *Stochastic finite elements: a spectral approach*. Courier Corporation.
- Glorot, X., & Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics* (pp. 249–256).
- Glorot, X., Bordes, A., & Bengio, Y. (2011). Deep sparse rectifier neural networks. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics* (pp. 315–323).
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT press.
- Gopalakrishnan, K., Khaitan, S. K., Choudhary, A., & Agrawal, A. (2017). Deep convolutional neural networks with transfer learning for computer vision-based data-driven pavement distress detection. *Construction and Building Materials*, 157, 322–330.
- Gurley, K., & Kareem, A. (1999). Applications of wavelet transforms in earthquake, wind and ocean engineering. *Engineering structures*, 21(2), 149–167.
- He, K., Zhang, X., Ren, S., & Sun, J. (2016a). Deep residual learning for image recognition. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 770–778).
- He, K., Zhang, X., Ren, S., & Sun, J. (2016b). Identity mappings in deep residual networks. In *European conference on computer vision* (pp. 630–645).
- Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). Densely connected convolutional networks. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 4700–4708).
- Huang, Y., Beck, J. L., & Li, H. (2018). Multitask sparse bayesian learning with applications in structural health monitoring. *Computer-Aided Civil and Infrastructure Engineering*.
- Hughes, T. J. (2012). *The finite element method: linear static and dynamic finite element analysis*. Courier Corporation.
- Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*.
- Kareem, A. (2008). Numerical simulation of wind effects: a probabilistic perspective. *Journal of Wind Engineering and Industrial Aerodynamics*, 96(10-11), 1472–1497.
- Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (pp. 1097–1105).
- LeCun, Y., Bottou, L., Bengio, Y., Haffner, P., et al. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278–2324.
- Le Maître, O., & Knio, O. M. (2010). *Spectral methods for uncertainty quantification: with applications to computational fluid dynamics*. Springer Science & Business Media.
- Li, L., Jamieson, K., DeSalvo, G., Rostamizadeh, A., & Talwalkar, A. (2016). Hyperband: A novel bandit-based approach to hyperparameter optimization. *arXiv preprint arXiv:1603.06560*.
- Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 3431–3440).
- Mindlin, R. D. (1951). Influence of rotatory inertia and shear on flexural motions of isotropic, elastic plates. *J. appl. Mech.*, 18, 31–38.
- Mo, S., Zhu, Y., Zabaras, N., Shi, X., & Wu, J. (2019). Deep convolutional encoder-decoder networks for uncertainty quantification of dynamic multiphase flow in heterogeneous media. *Water Resources Research*, 55(1), 703–728.
- Murphy, K. P. (2012). *Machine learning: a probabilistic perspective*. MIT press.
- Nabian, M. A., & Meidani, H. (2018). Deep learning for accelerated seismic reliability analysis of transportation networks. *Computer-Aided Civil and Infrastructure Engineering*, 33(6), 443–458.
- Odena, A., Dumoulin, V., & Olah, C. (2016). Deconvolution and checkerboard artifacts. *Distill*, 1(10), e3.
- Rafiei, M. H., & Adeli, H. (2017). A novel machine learning-based algorithm to detect damage in high-rise building structures. *The Structural Design of Tall and Special Buildings*, 26(18), e1400.
- Rafiei, M. H., & Adeli, H. (2018). A novel unsupervised deep learning model for global and local health condition assessment of structures. *Engineering Structures*, 156, 598–607.

- Raissi, M., Perdikaris, P., & Karniadakis, G. E. (2019). Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378, 686–707.
- Reissner, E. (1945). The effect of transverse shear deformation on the bending of elastic plates. *J. appl. Mech.*, A69–A77.
- Robbins, H., & Monro, S. (1951). A stochastic approximation method. *The annals of mathematical statistics*, 400–407.
- Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *International conference on medical image computing and computer-assisted intervention* (pp. 234–241).
- Rumelhart, D. E., Hinton, G. E., Williams, R. J., et al. (1988). Learning representations by back-propagating errors. *Cognitive modeling*, 5(3), 1.
- Sadeghi, N., Fayek, A. R., & Pedrycz, W. (2010). Fuzzy monte carlo simulation and risk assessment in construction. *Computer-Aided Civil and Infrastructure Engineering*, 25(4), 238–252.
- Schwab, C., & Gittelson, C. J. (2011). Sparse tensor discretizations of high-dimensional parametric and stochastic pdes. *Acta Numerica*, 20, 291–467.
- Shinozuka, M., & Jan, C.-M. (1972). Digital simulation of random processes and its applications. *Journal of sound and vibration*, 25(1), 111–128.
- Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.
- Sirignano, J., & Spiliopoulos, K. (2018). Dgm: A deep learning algorithm for solving partial differential equations. *Journal of Computational Physics*, 375, 1339–1364.
- Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1), 1929–1958.
- Viana, F. A., Simpson, T. W., Balabanov, V., & Toropov, V. (2014). Special section on multidisciplinary design optimization: metamodeling in multidisciplinary design optimization: how far have we really come? *AIAA journal*, 52(4), 670–690.
- Xiu, D., & Karniadakis, G. E. (2002). The wiener–askey polynomial chaos for stochastic differential equations. *SIAM journal on scientific computing*, 24(2), 619–644.
- Yang, X., Li, H., Yu, Y., Luo, X., Huang, T., & Yang, X. (2018). Automatic pixel-level crack detection and measurement using fully convolutional network. *Computer-Aided Civil and Infrastructure Engineering*, 33(12), 1090–1109.
- Yeum, C. M., & Dyke, S. J. (2015). Vision-based automated crack detection for bridge inspection. *Computer-Aided Civil and Infrastructure Engineering*, 30(10), 759–770.
- Zhu, Y., & Zabaras, N. (2018). Bayesian deep convolutional encoder–decoder networks for surrogate modeling and uncertainty quantification. *Journal of Computational Physics*, 366, 415–447.

