



Prediction of composite microstructure stress-strain curves using convolutional neural networks



Charles Yang^{a,1}, Youngsoo Kim^{b,1}, Seunghwa Ryu^{b,*}, Grace X. Gu^{a,*}

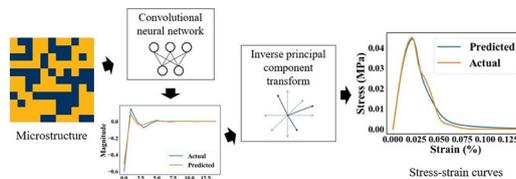
^a Department of Mechanical Engineering, University of California, Berkeley, CA 94720, USA

^b Department of Mechanical Engineering & KI for the NanoCentury, Korea Advanced Institute of Science and Technology, Daejeon 34141, Republic of Korea

HIGHLIGHTS

- Used convolutional neural networks to predict the entire stress-strain curve of binary composites with excellent accuracy
- Demonstrated the ability to compress stress-strain curves with negligible loss in information
- Established novel custom loss function for neural networks predicting compressed stress-strain curves
- Formulated interpretable evaluation metric of a machine learning model's ability to predict stress-strain curves

GRAPHICAL ABSTRACT



ARTICLE INFO

Article history:

Received 12 December 2019

Received in revised form 18 January 2020

Accepted 18 January 2020

Available online 20 January 2020

Keywords:

Machine learning
Convolutional neural networks
Mechanical properties
Microstructure
Computational mechanics

ABSTRACT

Stress-strain curves are an important representation of a material's mechanical properties, from which important properties such as elastic modulus, strength, and toughness, are defined. However, generating stress-strain curves from numerical methods such as finite element method (FEM) is computationally intensive, especially when considering the entire failure path for a material. As a result, it is difficult to perform high throughput computational design of materials with large design spaces, especially when considering mechanical responses beyond the elastic limit. In this work, a combination of principal component analysis (PCA) and convolutional neural networks (CNN) are used to predict the entire stress-strain behavior of binary composites evaluated over the entire failure path, motivated by the significantly faster inference speed of empirical models. We show that PCA transforms the stress-strain curves into an effective latent space by visualizing the eigenbasis of PCA. Despite having a dataset of only 10^{-27} % of possible microstructure configurations, the mean absolute error of the prediction is $<10\%$ of the range of values in the dataset, when measuring model performance based on derived material descriptors, such as modulus, strength, and toughness. Our study demonstrates the potential to use machine learning to accelerate material design, characterization, and optimization.

© 2020 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Understanding the relationship between structure and property for materials is a seminal problem in material science, with significant applications for designing next-generation materials. A primary motivating example is designing composite microstructures for load-bearing applications, as composites offer advantageously high specific strength

* Corresponding authors.

E-mail addresses: ryush@kaist.ac.kr (S. Ryu), ggxu@berkeley.edu (G.X. Gu).

¹ These authors contributed equally to this work.

and specific toughness. Recent advancements in additive manufacturing have facilitated the fabrication of complex composite structures, and as a result, a variety of complex designs have been fabricated and tested via 3D-printing methods [1–10]. While more advanced manufacturing techniques are opening up unprecedented opportunities for advanced materials and novel functionalities, identifying microstructures with desirable properties is a difficult optimization problem.

One method of identifying optimal composite designs is by constructing analytical theories. For conventional particulate/fiber-reinforced composites, a variety of homogenization theories have been developed to predict the mechanical properties of composites as a function of volume fraction, aspect ratio, and orientation distribution of reinforcements [2,11,12]. Because many natural composites, synthesized via self-assembly processes, have relatively periodic and regular structures, their mechanical properties can be predicted if the load transfer mechanism of a representative unit cell and the role of the self-similar hierarchical structure are understood [13]. However, the applicability of analytical theories is limited in quantitatively predicting composite properties beyond the elastic limit in the presence of defects, because such theories rely on the concept of representative volume element (RVE), a statistical representation of material properties, whereas the strength and failure is determined by the weakest defect in the entire sample domain. Numerical modeling based on finite

element methods (FEM) can complement analytical methods for predicting inelastic properties such as strength and toughness modulus (referred to as toughness, hereafter) which can only be obtained from full stress-strain curves.

However, numerical schemes capable of modeling the initiation and propagation of the curvilinear cracks, such as the crack phase field model, are computationally expensive and time-consuming because a very fine mesh is required to accommodate highly concentrated stress field near crack tip and the rapid variation of damage parameter near diffusive crack surface [14–21]. Meanwhile, analytical models require significant human effort and domain expertise and fail to generalize to similar domain problems. In order to identify high-performing composites in the midst of large design spaces within realistic time-frames, we need models that can rapidly describe the mechanical properties of complex systems and be generalized easily to analogous systems. Machine learning offers the benefit of extremely fast inference times and requires only training data to learn relationships between inputs and outputs e.g., composite microstructures and their mechanical properties. Machine learning has already been applied to speed up the optimization of several different physical systems, including graphene kirigami cuts [22], fine-tuning spin qubit parameters [23], and probe microscopy tuning [24]. Such models do not require significant human intervention or knowledge, learn relationships efficiently relative to

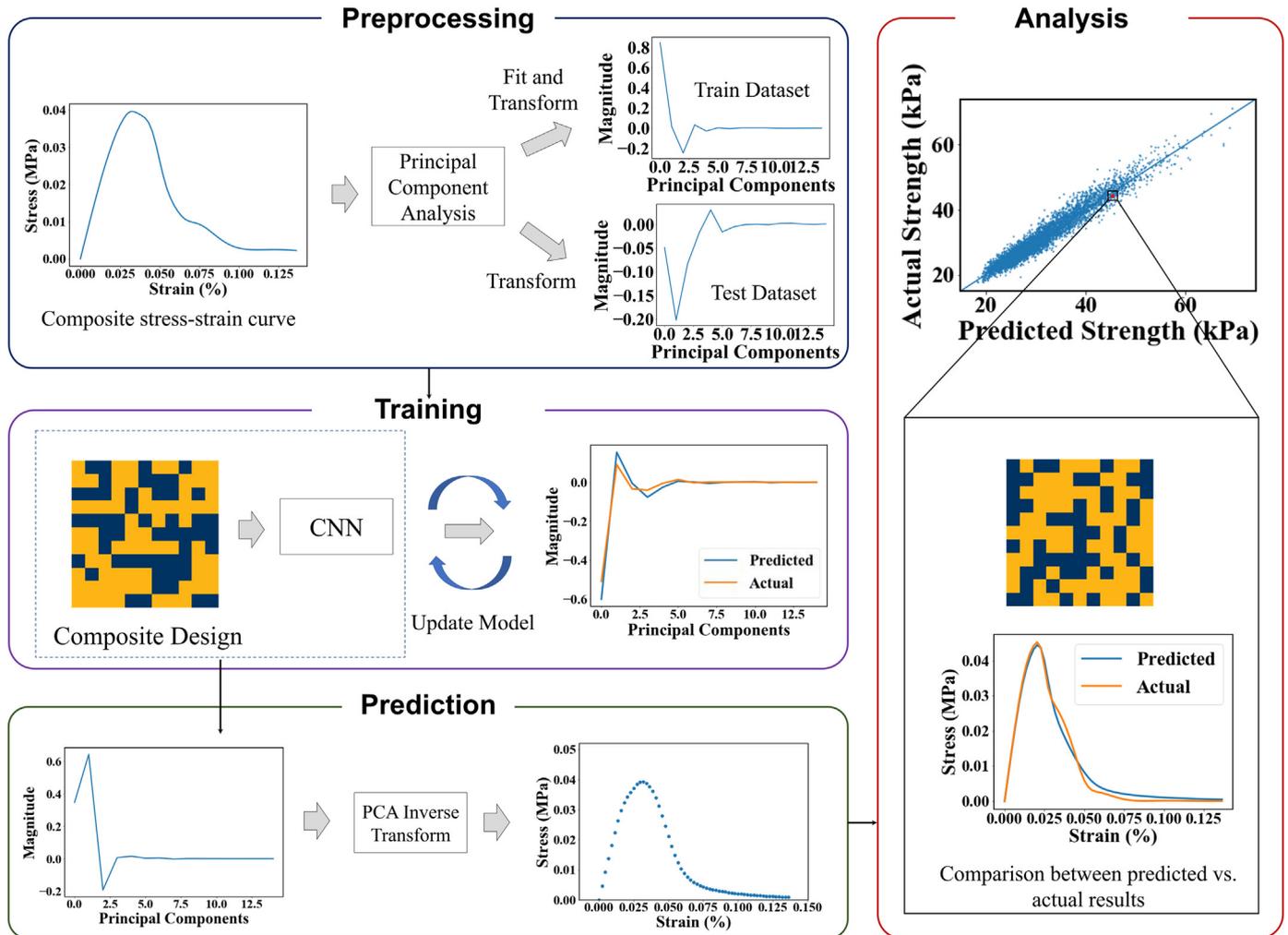


Fig. 1. Workflow of modeling process. To preprocess stress-strain curves, we convert them to PCA-coordinate representations, fitting a PCA model to the training set and applying the learned model to compress both the training and test set. Our CNN model takes as input the composite design and makes predictions in the PCA-coordinate space, with the loss function weighted by the explained variance ratio of each component. Finally, the CNN makes predictions about composite designs it hasn't seen. These PCA predictions are transformed back into regular stress-strain curves. To analyze our results, we compare the predicted material descriptors compared to the actual ones, derived from the predicted and actual stress-strain curves.

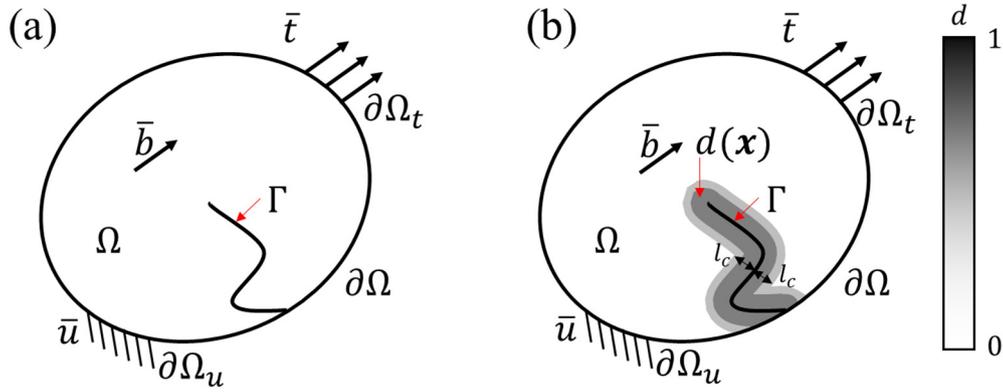


Fig. 2. Schematic of two-dimensional crack topology: (a) sharp crack model and (b) diffusive crack model described by phase field $d(x)$.

the input design space, and can be generalized to different systems [25–34].

In this paper, we utilize a combination of principal component analysis (PCA) and convolutional neural networks (CNN) to predict the entire stress-strain curve of composite failures beyond the elastic limit. Stress-strain curves are chosen as the model’s target because they are difficult to predict given their high dimensionality. In addition, stress-strain curves are used to derive important material descriptors such as modulus, strength, and toughness. In this sense, predicting stress-strain curves is a more general description of composites properties than any combination of scalar material

descriptors. A dataset of 100,000 different composite microstructures and their corresponding stress-strain curves are used to train and evaluate model performance. Due to the high dimensionality of the stress-strain dataset, several dimensionality reduction methods are used, including PCA, featuring a blend of domain understanding and traditional machine learning, to simplify the problem without loss of generality for the model.

We will first describe our modeling methodology and the parameters of our finite-element method (FEM) used to generate data. Visualizations of the learned PCA latent space are then presented, along with model performance results.

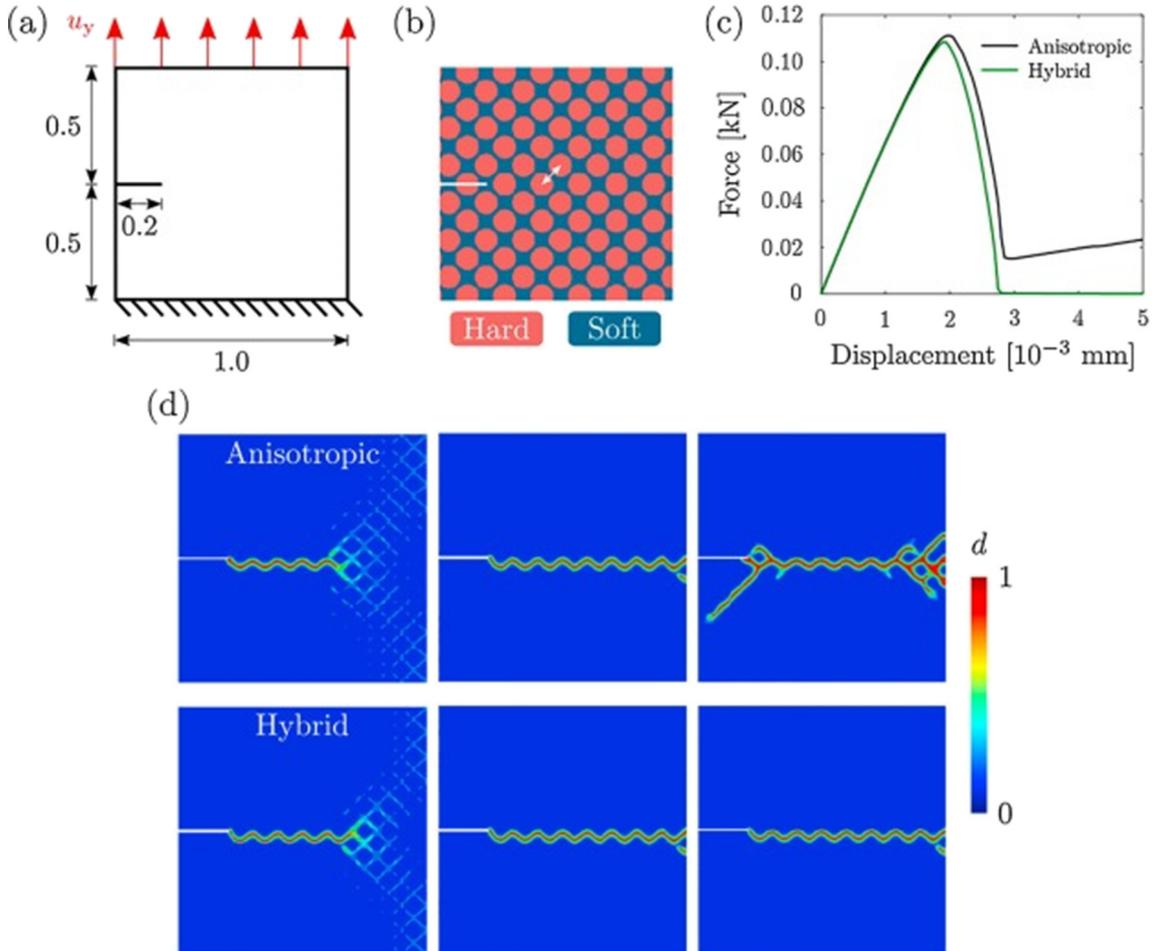


Fig. 3. (a) Geometry and boundary condition for single-edge notched specimen and (b) its microstructure composed of two different materials with high stiffness ratio. (c) Force-displacement curves under tension and (d) crack patterns obtained from anisotropic and hybrid formulations [15].

2. Method

2.1. Modeling stress-strain curves with PCA

We represent all the stress-strain curves as arrays of stress values evaluated at 61 strain points (referred to as stress vector, hereafter), and reduce the dimensionality of the stress vector by employing PCA, a linear method that identifies an orthonormal basis along which to project the data in a lower dimension. PCA is only fit on training data to find the orthonormal basis, which is defined by principal component vectors. Using this basis, both the training data and test data are transformed by projecting the stress vector into the lower dimensional latent space and finding a new set of coordinates to define the stress vectors.

$$\bar{\mathbf{X}} = \mathbf{X} - \bar{\boldsymbol{\mu}} \quad (1)$$

$$\bar{\mathbf{X}} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T \quad (2)$$

The $100,000 \times 61$ stress matrix is denoted as the \mathbf{X} matrix, where each row represents a stress-strain curve and each column a stress value at a given strain value. Even though stress-strain curves are composed of a stress and a strain vector, only the stress vector is passed to PCA, because the strain vector across all composite blocks had the same values and lengths. Using the strain vector would have added no extra information to the model because it is constant for all inputs. This allows us to significantly reduce the dimensionality of the output, making it easier for the model to make predictions. The stress matrix is first mean-centered by column as shown in Eq. (1), so that each column has a mean of 0. PCA utilizes singular value decomposition (SVD), shown in Eq. (2), which is a real matrix factorization method. The $\boldsymbol{\Sigma}^T\boldsymbol{\Sigma}$ matrix contains the eigenvalues that determine the variance explained in each principal component while the \mathbf{V}^T matrix contains the principal components in order of decreasing variance explained. The first 15 principal components were chosen to be kept and the rest discarded. This is done by projecting the data along the first 15 principal components in \mathbf{V}^T to obtain a 15-dimensional representation of the stress-strain curve. The cumulative explained variance (C.E.V.) is the sum of the first 15 eigenvalues normalized by the sum of all eigenvalues, shown in Eq. (3). By using dimensionality reduction, our CNN model is able to learn more efficiently because it is learning in a lower dimensional space, so that we can obtain higher accuracy with less training data.

$$C.E.V.(n = 15) = \frac{\sum_{i=1}^n \lambda_i}{\sum_{j=1}^{61} \lambda_j} \quad (3)$$

2.2. Creating a customized loss function for CNN

Each eigenvalue represents the explained variance of the associated principal component basis-vector. In order to communicate to our CNN that some principal components were more important than others, we constructed a customized loss function that weighted errors in each principal component by the associated eigenvalue i.e., principal components that have larger explained variance were weighted more heavily in the loss function than those with less explained variances. In Eq. (4), we show the standard mean squared error loss function compared to the PCA weighted loss function we devised, as shown in Eq. (5).

$$MSE = \sum_{i=1}^n \sum_{j=1}^{15} (y_i[j] - \hat{y}_i[j])^2 \quad (4)$$

$$Weighted\ MSE = \sum_{i=1}^n \sum_{j=1}^{15} \frac{\lambda_j}{\sum_{k=1}^{61} \lambda_k} (y_i[j] - \hat{y}_i[j])^2 \quad (5)$$

To give intuition regarding the formulation of this customized loss function, recall that the CNN now outputs a 15-dimensional vector as a prediction. However, each component in this 15-dimensional vector corresponds to a particular principal component coordinate, which has a corresponding eigenvalue denoting its importance. Our customized loss function in Eq. (5) uses the explained variance ratio of each component in the loss function to 'teach' the model that certain components matter more than others, whereas the naive mean squared error loss function in Eq. (4) treats each component equally. A flow diagram of the data processing, model training, and error analysis is shown in Fig. 1.

2.3. CNN implementation and training

A convolutional neural network was trained to predict this lower dimensional representation of the stress vector. The input to the CNN was a binary matrix representing the composite design, with 0's corresponding to soft blocks and 1's corresponding to stiff blocks. PCA was implemented with the open-source Python package scikit-learn [35], using the default hyperparameters. CNN was implemented using Keras [36] with a TensorFlow backend. The batch size for all experiments was set to 16 and the number of epochs to 30; the Adam optimizer was used to update the CNN weights during backpropagation [37].

A train/test split ratio of 95:5 is used – we justify using a smaller ratio than the standard 80:20 because of a relatively large dataset. With a ratio of 95:5 and a dataset with 100,000 instances, the test set size still has enough data points, roughly several thousands, for its results to generalize. Each column of the target PCA-representation was normalized to have a mean of 0 and a standard deviation of 1 to prevent unstable training.

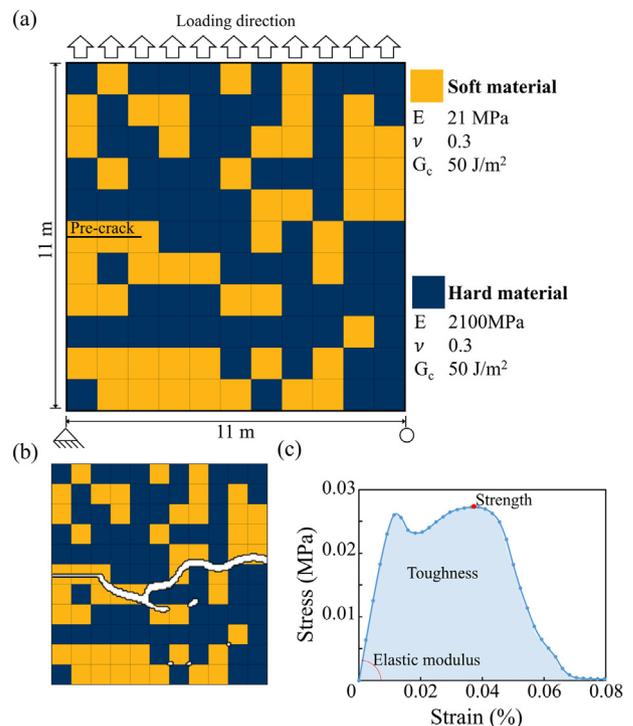


Fig. 4. (a) An example of composite plate and boundary conditions to test to the composites in this paper. Each composite plate is composed of 121 square blocks. 'E, ν, G_c' represent elastic modulus, Poisson's ratio and energy release rate, respectively (b) Totally fractured state based on crack phase field model. (c) Stress-strain curve and the graphical definitions of elastic modulus, strength and toughness.

2.4. Finite element method data generation

FEM was used to generate training data for the CNN model. Although initially obtained training data is compute-intensive, it takes much less time to train the CNN model and even less time to make high-throughput inferences over thousands of new, randomly generated composites. The crack phase field solver was based on the hybrid formulation for the quasi-static fracture of elastic solids and implemented in the commercial FEM software ABAQUS with a user-element subroutine (UEL).

The crack phase field model is a variational approach where the crack discontinuity is approximated as a continuous scalar field called crack phase field $d(x)$. The two different models for crack discontinuities are shown in Fig. 2 [18–20]. The phase parameter has a diffusive nature with a regularization parameter, l_c , and the diffusive crack converges to the original sharp crack in the limit as $l_c \rightarrow 0$. Crack phase field modeling allows one to simulate complex crack evolution such as curvilinear crack path, crack branching or coalescence. Also, unlike XFEM (extended-FEM) which requires predefined crack initiation site or path, entire fracture process involving crack nucleation and propagation can be

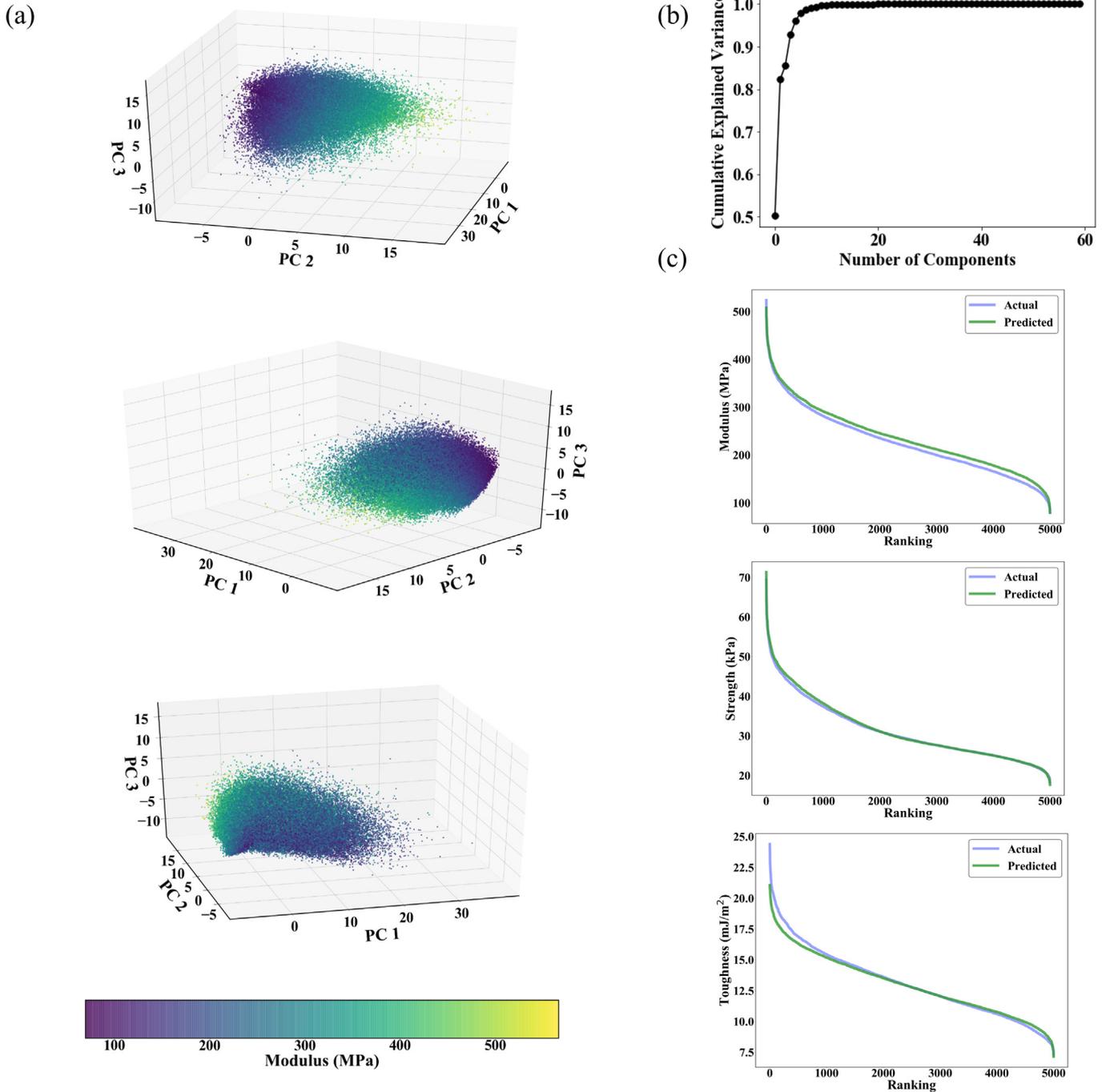


Fig. 5. (a) Scatterplots of stress-strain curves represented in the 3D latent space defined by the first 3 principal component vectors in PCA, which contain ~85% of the variance in the data. The color scale represents the associated modulus value of each stress-strain curve. The plot is viewed from different azimuthal angles. (b) Scree plot of PCA i.e., cumulative explained variance ratio with respect to number of dimensions. (c) Comparing the distribution of values learned based on ranking the values as derived from the convolutional neural network stress-strain predictions for modulus, strength, and toughness.

Table 1
Using a convolutional neural network and 15-dimensional PCA reduction, we evaluate the performance of our model over 15 trials with a 95:5 train test split. We measure performance of CNN at predicting stress-strain curves by the error between the actual and predicted material descriptor values (modulus, strength, and toughness) derived from stress-strain curves. 95% confidence intervals are included alongside with average values. Statistical descriptors about the distribution of values taken on by material descriptors are provided for context.

	Mean absolute error	Mean squared error	R ²	Max	Min	Range	Average
Modulus (MPa)	18.3 ± 6.16	533 ± 328	0.888 ± 0.0652	566.04	72.96	493.08	225.85
Strength (kPa)	1.66 ± 0.479	4.69 ± 2.61	0.918 ± 0.0455	71.67	17.22	54.45	31.16
Toughness (mj/m ²)	1.04 ± 0.106	1.86 ± 0.230	0.763 ± 0.0261	24.79	7.40	17.39	13.07

modeled based on a single criterion involving the critical energy release rate.

While there are several methods to formulate the crack phase field model, we employed the hybrid formulation [14] which has been reported to be adequate for modeling the fracture of composite materials [15]. In the hybrid formulation, the key idea is to couple the crack phase field $d(\mathbf{x})$ with the displacement field $\mathbf{u}(\mathbf{x})$ via the degradation of strain energy density ψ as shown in Eq. (6),

$$\psi(\epsilon(\mathbf{u}), d) = (1-d)^2 \psi_0^+(\epsilon) + \psi_0^-(\epsilon) \quad (6)$$

where ϵ , ψ_0^+ , and ψ_0^- refer to strain tensor, tensile part of strain energy

density and compressive part of strain energy density obtained without considering $d(\mathbf{x})$, respectively. Eq. (7) shows how we can link the crack phase field $d(\mathbf{x})$ with the stress field $\sigma(\mathbf{x})$ via the degradation of elastic stiffness,

$$\begin{aligned} \sigma(\mathbf{u}, d) &= [(1-d)^2 + k] \sigma_0 = [(1-d)^2 + k] \mathbf{C}_0 \epsilon \\ &= [(1-d)^2 + k] \frac{\partial \psi_0(\epsilon)}{\partial \epsilon} \end{aligned} \quad (7)$$

where σ refers to stress tensor accounting for $d(\mathbf{x})$ while σ_0 and \mathbf{C}_0 refers to stress tensor and stiffness tensor without considering $d(\mathbf{x})$. k is a very small dummy constant which is inserted to ensure numerical

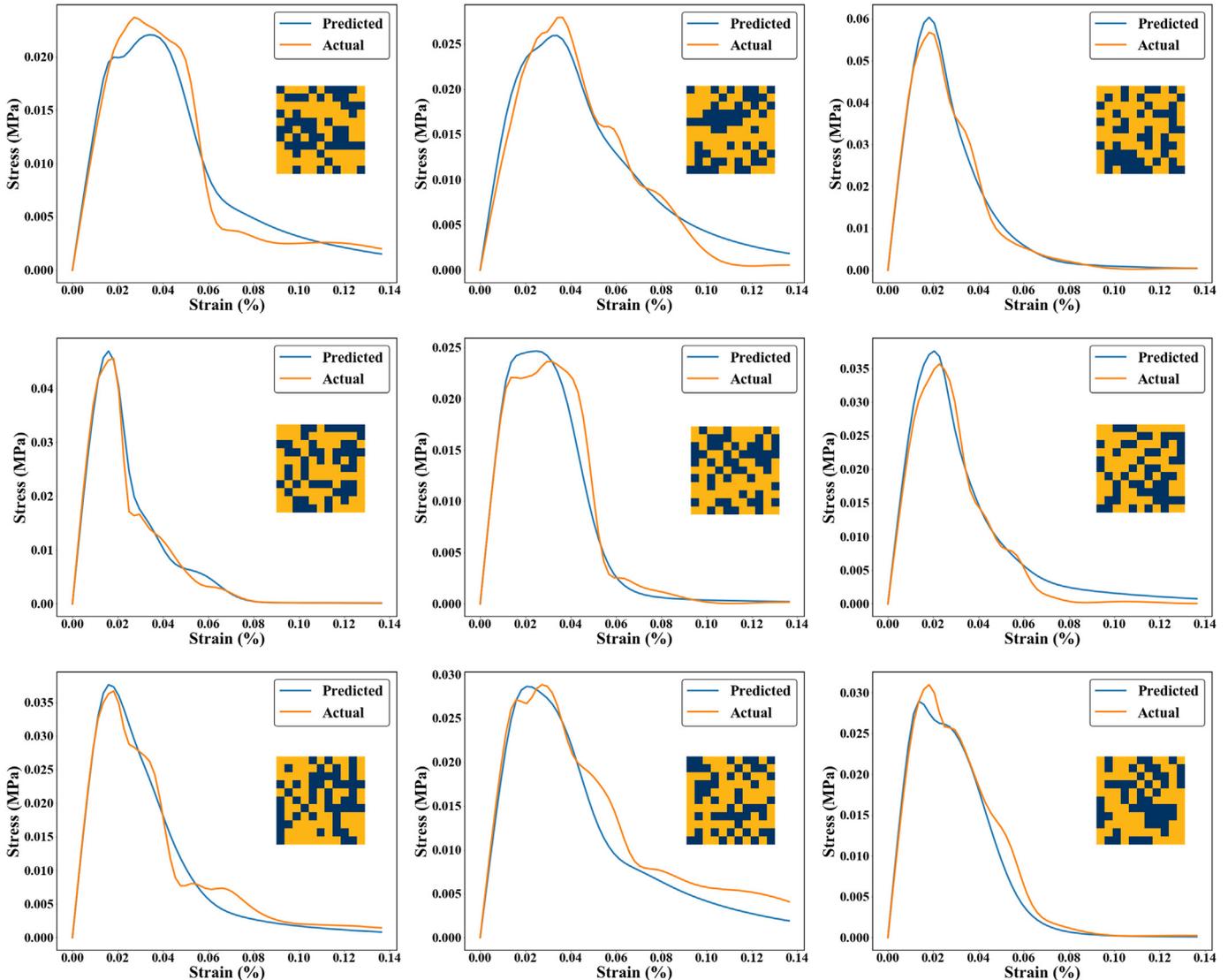


Fig. 6. A randomly chosen set of 9 predicted stress-strain curves compared to the actual stress-strain curve with the associated composite included. These examples, which are drawn from the test set, have mean squared errors in the 25–75% of the mean squared error distribution for the test set, providing a representative sample of examples.

convergence. The load bearing capacity disappears when d goes to zero. Since the stiffness degradation equation indicates zero load bearing capacity under any loading condition, an additional constraint is necessary to prevent crack face inter-penetration under compression, as in

$$\forall \mathbf{x} : \text{if } \psi_0^+ \mathbf{b} \psi_0^- \rightarrow d = 0 \quad (8)$$

The crack phase field is updated via the critical energy release rate criterion:

$$d - l^2 \Delta d = \frac{2l}{g_c} (1-d) H^+ \quad \text{in } \Omega$$

$$\nabla d \cdot \mathbf{n} = 0 \quad \text{on } \Gamma, \partial \Omega$$

where irreversibility condition is ensured by history variable $H^+ = \max_{\tau \in [0, t]}$

$\psi_0^+(\boldsymbol{\epsilon}(\mathbf{x}, \tau))$ (i.e. crack healing is prohibited).

The critical advantage of hybrid formulation over the widely used anisotropic formulation [18] lies in the appropriate description of the stiffness degradation. The stiffness degradation in the anisotropic formulation is given by the following:

$$\boldsymbol{\sigma}(\mathbf{u}, d) = \left[(1-d)^2 + k \right] \frac{\partial \psi_0^+(\boldsymbol{\epsilon})}{\partial \boldsymbol{\epsilon}} + \frac{\partial \psi_0^-(\boldsymbol{\epsilon})}{\partial \boldsymbol{\epsilon}} \quad (10)$$

This stiffness degradation scheme, which exclusively considers the tensile part of the strain energy density, leads to spurious load bearing capacity upon combined tensile and shear loading near the crack tip. As shown in Fig. 3, the anisotropic formulation predicts spurious crack growth and unphysical load-displacement response after complete fracture, while the hybrid formulation offers reasonable fracture pattern and load-displacement curve. Because full stress-strain responses of composites are considered, the hybrid formulation is chosen in the present study.

FEM simulations are performed with 2D plane stress conditions with CPS4 element of a large plate divided into 121 blocks, with 70 stiff and 51 soft blocks randomly assigned to 121 sites in the composite unit cell, resulting in $>10^{34}$ distinct configurations. Stress-strain curves of 100,000 composite configurations are obtained until the complete fracture with strain increment of 0.0000227 while the interface between stiff and soft blocks is assumed to be perfectly bonded. Each block was divided into 144 elements (12 by 12) and the regularization parameter, i.e. characteristic diffusive length of the crack, is set to be twice the length of each element. All configurations fail completely at the maximum strain of 0.00136. Hence, each stress-strain curve can be represented as an array of 61 stress values. See Fig. 4 for a description of the FEM model inputs and outputs.

3. Results and discussion

3.1. Visualizing PCA

In order to better understand the role PCA plays in effectively capturing the information contained in stress-strain curves, the principal component representation of stress-strain curves is plotted in 3 dimensions. Specifically, we take the first three principal components, which have a cumulative explained variance ~85%, and plot stress-strain curves in that basis in Fig. 5a and provide several different angles from which to view the 3D plot. Each point represents a stress-strain curve in the PCA latent space and is colored based on the associated modulus value. Based on Fig. 5a, it seems that the PCA is able to spread out the curves in the latent space based on modulus values, which suggests that this is a useful latent space for CNN to make predictions in.

The cumulative explained variance ratio with respect to the number of dimensions is shown in Fig. 5b. As shown in the figure we are able to capture >99.5% of the variance in the stress-strain curves, a factor of 4

reduction in dimensionality with negligible loss in quality of information given to the model by using only 15 principal components. While it is difficult to visualize stress-strain curves as 61-dimensional vectors, we can intuitively understand why PCA is able to find a linear lower dimensional representation based on our understanding of stress-strain curves. Because these curves have a linear portion i.e. elastic deformation, followed by fracture and crack propagation, they have generally similar.

3.2. CNN model design and performance

Our CNN was a fully convolutional neural network i.e. the only dense layer was the output layer. All convolution layers used 16 filters with a stride of 1, with a LeakyReLU activation followed by BatchNormalization. The first 3 Conv blocks did not have 2D MaxPooling, followed by 9 conv blocks which did have a 2D MaxPooling layer, placed after the BatchNormalization layer. A GlobalAveragePooling was used to reduce the dimensionality of the output tensor from the sequential convolution blocks and the final output layer was a Dense layer with 15 nodes, where each node corresponded to a principal component. In total, our model had 26,319 trainable weights.

Our architecture was motivated by the recent development and convergence onto fully-convolutional architectures for traditional computer vision applications, where convolutions are empirically observed to be more efficient and stable for learning as opposed to dense layers. In addition, in our previous work, we had shown that CNN's were a capable architecture for learning to predict mechanical properties of 2D composites [30]. The convolution operation is an intuitively good fit for predicting crack propagation because it is a local operation, allowing it to implicitly featurize and learn the local spatial effects of crack propagation.

After applying PCA transformation to reduce the dimensionality of the target variable, CNN is used to predict the PCA representation of the stress-strain curve of a given binary composite design. After training the CNN on a training set, its ability to generalize to composite designs it has not seen is evaluated by comparing its predictions on an unseen test set. However, a natural question that emerges is how to evaluate a model's performance at predicting stress-strain curves in a real-world engineering context. While simple scalar metrics such as mean squared error (MSE) and mean absolute error (MAE) generalize easily to vector targets, it is not clear how to interpret these aggregate summaries of performance. It is difficult to use such metrics to ask questions such as "Is this model good enough to use in the real world" and "On average, how poorly will a given prediction be incorrect relative to some given specification". Although being able to predict stress-strain curves is an important application of FEM and a highly desirable property for any machine learning model to learn, it does not easily lend itself to interpretation. Specifically, there is no simple quantitative way to define whether two stress-strain curves are "close" or "similar" with real-world units.

Given that stress-strain curves are oftentimes intermediary representations of a composite property that are used to derive more meaningful descriptors such as modulus, strength, and toughness, we decided to evaluate the model in an analogous fashion. The CNN prediction in the PCA latent space representation is transformed back to a stress-strain curve using PCA, and used to derive the predicted modulus, strength, and toughness of the composite. The predicted material descriptors are then compared with the actual material descriptors. In this way, MSE and MAE now have clearly interpretable units and meanings. The average performance of the model with respect to the error between the actual and predicted material descriptor values derived from stress-strain curves are presented in Table 1. The MAE for material descriptors provides an easily interpretable metric of model performance and can easily be used in any design specification to provide confidence estimates of a model prediction. When comparing the mean

absolute error (MAE) to the range of values taken on by the distribution of material descriptors, we can see that the MAE is relatively small compared to the range. The MAE compared to the range is <10% for all material descriptors. Relatively tight confidence intervals on the error indicate that this model architecture is stable, the model performance is not heavily dependent on initialization, and that our results are robust to different train-test splits of the data.

Ranking plots are shown in Fig. 5c and demonstrate that the CNN can accurately learn the underlying distribution of values. A sample of predicted versus actual stress-strain curves are provided in Fig. 6. Despite complex non-smooth behaviors exhibited in stress-strain curves, the CNN is able to approximate the non-linear portions of the stress-strain curves well despite only using linear dimensionality reduction, which speaks to the learning ability of CNN to learn complex behaviors just from the microstructure. In addition, our CNN's ability to predict modulus, strength, and toughness values derived from stress-strain curves is significantly improved over our previous work [30], which directly predicted the scalar material properties from the composite design.

3.3. Future work

Future work includes combining empirical models with optimization algorithms, such as gradient-based methods, to identify composite designs that yield complementary mechanical properties. The ability of a trained empirical model to make high-throughput predictions over designs it has never seen before allows for large parameter space optimization that would be computationally infeasible for FEM. In addition, we plan to explore different visualizations of empirical models in an effort to "open up the black-box" of such models. Applying machine learning to finite-element methods is a rapidly growing field with the potential to discover novel next-generation materials tailored for a variety of applications. We also note that the proposed method can be readily applied to predict other physical properties represented in a similar vectorized format, such as electron/phonon density of states, and sound/light absorption spectrum.

4. Conclusion

In conclusion, we applied PCA and CNN to rapidly and accurately predict the stress-strain curves of composites beyond the elastic limit. In doing so, several novel methodological approaches were developed, including using the derived material descriptors from the stress-strain curves as interpretable metrics for model performance and dimensionality reduction techniques to stress-strain curves. This method has the potential to enable composite design with respect to mechanical response beyond the elastic limit, which was previously computationally infeasible, and can generalize easily to related problems outside of microstructural design for enhancing mechanical properties.

CRedit authorship contribution statement

Charles Yang: Writing - original draft, Writing - review & editing, Conceptualization, Formal analysis. **Youngsoo Kim:** Writing - original draft, Writing - review & editing, Conceptualization, Formal analysis. **Seunghwa Ryu:** Writing - original draft, Writing - review & editing, Conceptualization, Formal analysis. **Grace X. Gu:** Writing - original draft, Writing - review & editing, Conceptualization, Formal analysis.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The authors acknowledge support from the Extreme Science and Engineering Discovery Environment (XSEDE) Bridges system at the Pittsburgh Supercomputing Center (PSC), which is supported by National Science Foundation grant number ACI-1548562. The authors also acknowledge support from an NVIDIA GPU Seed Grant and the Savio computational cluster resource provided by the Berkeley Research Computing program. Additionally, the authors acknowledge support by the Basic Science Research Program (2019R1A2C4070690) and Creative Materials Discovery Program (2016M3D1A1900038) through the National Research Foundation of Korea (NRF), as well as the support by the KAIST-funded Global Singularity Research Program for 2019 (N1190118).

Data availability

The training data for the study is available upon request.

References

- [1] A. Ghazlan, T.D. Ngo, P. Tran, Three-dimensional Voronoi model of a nacre-mimetic composite structure under impulsive loading, *Compos. Struct.* 153 (2016) 278–296, <https://doi.org/10.1016/j.compstruct.2016.06.020>.
- [2] Y. Kim, Y. Kim, T.I. Lee, T.S. Kim, S. Ryu, An extended analytic model for the elastic properties of platelet-staggered composites and its application to 3D printed structures, *Compos. Struct.* 189 (2018) 27–36, <https://doi.org/10.1016/j.compstruct.2018.01.038>.
- [3] N. Sakhavand, R. Shahsavari, Universal composition–structure–property maps for natural and biomimetic platelet–matrix composites and stacked heterostructures, *Nat. Commun.* 6 (2015) 6523, <https://doi.org/10.1038/ncomms7523>.
- [4] P. Tran, T.D. Ngo, A. Ghazlan, D. Hui, Bimaterial 3D printing and numerical analysis of bio-inspired composite structures under in-plane and transverse loadings, *Compos. Part B Eng.* 108 (2017) 210–223, <https://doi.org/10.1016/j.compositesb.2016.09.083>.
- [5] P. Zhang, M.A. Heyne, A.C. To, Biomimetic staggered composites with highly enhanced energy dissipation: modeling, 3D printing, and testing, *J. Mech. Phys. Solids* 83 (2015) 285–300, <https://doi.org/10.1016/j.jmps.2015.06.015>.
- [6] Z. Zhang, K.G. Demir, G.X. Gu, Developments in 4D-printing: a review on current smart materials, technologies, and applications, *Int. J. Smart Nano Mater.* 00 (2019) 1–20, <https://doi.org/10.1080/19475411.2019.1591541>.
- [7] B.G. Compton, J.A. Lewis, 3D-printing of lightweight cellular composites, *Adv. Mater.* 26 (2014) 5930–5935, <https://doi.org/10.1002/adma.201401804>.
- [8] S. Das, D.L. Bourell, S.S. Babu, Metallic materials for 3D printing, *MRS Bull.* 41 (2016) 729–741, <https://doi.org/10.1557/mrs.2016.217>.
- [9] B.H. Jared, M.A. Aguilo, L.L. Beghini, B.L. Boyce, B.W. Clark, A. Cook, B.J. Kaehr, J. Robbins, Additive manufacturing: toward holistic design, *Scr. Mater.* 135 (2017) 141–147, <https://doi.org/10.1016/j.scriptamat.2017.02.029>.
- [10] K. Makiewicz, M. Keller, A. Chaudhary, S. Babu, Microstructure evolution during laser additive manufacturing of Ti6Al4V alloys, *ASM Proc. Int. Conf. Trends Weld. Res.* 2012.
- [11] S. Lee, S. Ryu, Theoretical study of the effective modulus of a composite considering the orientation distribution of the fillers and the interfacial damage, *Eur. J. Mech. A. Solids* 72 (2018) 79–87, <https://doi.org/10.1016/j.euromechsol.2018.02.008>.
- [12] S. Ryu, S. Lee, J. Jung, J. Lee, Y. Kim, Micromechanics-based homogenization of the effective physical properties of composites with an anisotropic matrix and interfacial imperfections, *Front. Mater.* 6 (2019) 21, <https://doi.org/10.3389/fmats.2019.00021>.
- [13] J. Sun, B. Bhushan, Hierarchical structure and mechanical properties of nacre: a review, *RSC Adv.* 2 (2012) 7617–7632, <https://doi.org/10.1039/C2RA20218B>.
- [14] M. Ambati, T. Gerasimov, L. De Lorenzis, A review on phase-field models of brittle fracture and a new fast hybrid formulation, *Comput. Mech.* 55 (2015) 383–405, <https://doi.org/10.1007/s00466-014-1109-y>.
- [15] H. Jeong, S. Signetti, T.S. Han, S. Ryu, Phase field modeling of crack propagation under combined shear and tensile loading with hybrid formulation, *Comput. Mater. Sci.* 155 (2018) 483–492, <https://doi.org/10.1016/j.commatsci.2018.09.021>.
- [16] Y. Kim, H. Jeong, G.X. Gu, S. Ryu, A three-dimensional fracture pattern diagram of staggered platelet structures, *Compos. Struct.* 220 (2019) 769–775, <https://doi.org/10.1016/j.compstruct.2019.04.017>.
- [17] G. Liu, Q. Li, M.A. Msekh, Z. Zuo, Abaqus implementation of monolithic and staggered schemes for quasi-static and dynamic fracture phase-field model, *Comput. Mater. Sci.* 121 (2016) 35–47, <https://doi.org/10.1016/j.commatsci.2016.04.009>.
- [18] C. Miehe, M. Hofacker, F. Welschinger, A phase field model for rate-independent crack propagation: robust algorithmic implementation based on operator splits, *Comput. Methods Appl. Mech. Eng.* 199 (2010) 2765–2778, <https://doi.org/10.1016/j.cma.2010.04.011>.
- [19] C. Miehe, F. Welschinger, M. Hofacker, Thermodynamically consistent phase-field models of fracture: variational principles and multi-field FE implementations, *Int. J. Numer. Methods Eng.* 83 (2010) 1273–1311, <https://doi.org/10.1002/nme.2861>.

- [20] G. Molnár, A. Gravouil, 2D and 3D Abaqus implementation of a robust staggered phase-field solution for modeling brittle fracture, *Finite Elem. Anal. Des.* 130 (2017) 27–38, <https://doi.org/10.1016/j.finel.2017.03.002>.
- [21] T.T. Nguyen, J. Yvonnet, Q.-Z. Zhu, M. Bornert, C. Chateau, A phase field method to simulate crack nucleation and propagation in strongly heterogeneous materials from direct imaging of their microstructure, *Eng. Fract. Mech.* 139 (2015) 18–39, <https://doi.org/10.1016/j.engfracmech.2015.03.045>.
- [22] P.Z. Hanakata, E.D. Cubuk, D.K. Campbell, H.S. Park, Accelerated search and design of stretchable graphene kirigami using machine learning, *Phys. Rev. Lett.* 121 (2018), 255304. <https://doi.org/10.1103/PhysRevLett.121.255304>.
- [23] J.D. Teske, S.S. Humpohl, R. Otten, P. Bethke, P. Cerfontaine, J. Dedden, A. Ludwig, A.D. Wieck, H. Bluhm, A machine learning approach for automated fine-tuning of semiconductor spin qubits, *Appl. Phys. Lett.* 114 (2019), 133102. <https://doi.org/10.1063/1.5088412>.
- [24] R.A.J. Woolley, J. Stirling, A. Radocea, N. Krasnogor, P. Moriarty, Automated probe microscopy via evolutionary optimization at the atomic scale, *Appl. Phys. Lett.* 98 (2011), 253104. <https://doi.org/10.1063/1.3600662>.
- [25] N. Du, Z. Yang, X.Y. Liu, Y. Li, H.Y. Xu, Structural origin of the strain-hardening of spider silk, *Adv. Funct. Mater.* 21 (2011) 772–778, <https://doi.org/10.1002/adfm.201001397>.
- [26] G.X. Gu, C.T. Chen, D.J. Richmond, M.J. Buehler, Bioinspired hierarchical composite design using machine learning: simulation, additive manufacturing, and experiment, *Mater. Horizons* 5 (2018) 939–945, <https://doi.org/10.1039/c8mh00653a>.
- [27] G. Montavon, M. Rupp, V. Gobre, A. Vazquez-Mayagoitia, K. Hansen, A. Tkatchenko, K.-R. Müller, O. Anatole von Lilienfeld, Machine learning of molecular electronic properties in chemical compound space, *New J. Phys.* 15 (2013), 95003. <https://doi.org/10.1088/1367-2630/15/9/095003>.
- [28] G. Pilania, C. Wang, X. Jiang, S. Rajasekaran, R. Ramprasad, Accelerating materials property predictions using machine learning, *Sci. Rep.* 3 (2013) 2810, <https://doi.org/10.1038/srep02810>.
- [29] C.-T. Chen, G.X. Gu, Machine learning for composite materials, *MRS Commun.* 9 (2019) 556–566, <https://doi.org/10.1557/mrc.2019.32>.
- [30] C. Yang, Y. Kim, S. Ryu, G.X. Gu, Using convolutional neural networks to predict composite properties beyond the elastic limit, *MRS Commun.* 9 (2019) 609–617, <https://doi.org/10.1557/mrc.2019.49>.
- [31] G.B. Goh, N.O. Hodas, A. Vishnu, Deep learning for computational chemistry, *J. Comput. Chem.* 38 (2017) 1291–1307, <https://doi.org/10.1002/jcc.24764>.
- [32] Y. Lecun, Y. Bengio, G. Hinton, Deep learning, *Nature* 521 (2015) 436–444, <https://doi.org/10.1038/nature14539>.
- [33] L. Li, Y. Yang, D. Zhang, Z.-G. Ye, S. Jesse, S.V. Kalinin, R.K. Vasudevan, Machine learning-enabled identification of material phase transitions based on experimental data: exploring collective dynamics in ferroelectric relaxors, *Sci. Adv.* 4 (2018) <https://doi.org/10.1126/sciadv.aap8672>.
- [34] S.M. Neumayer, L. Collins, R. Vasudevan, C. Smith, S. Somnath, V.Y. Shur, S. Jesse, A.L. Kholkin, S.V. Kalinin, B.J. Rodriguez, Decoupling mesoscale functional response in PLZT across the ferroelectric-relaxor phase transition with contact kelvin probe force microscopy and machine learning, *ACS Appl. Mater. Interfaces* 10 (2018) 42674–42680, <https://doi.org/10.1021/acsami.8b15872>.
- [35] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: machine learning in Python, *J. Mach. Learn. Res.* 12 (2011) 2825–2830, <https://doi.org/10.1007/s13398-014-0173-7.2>.
- [36] F. Chollet, Keras, Github <https://github.com/keras-team/keras> 2015.
- [37] D.P. Kingma, J. Ba, Adam: a method for stochastic optimization, *CoRR. Abs/1412 (2015) 6*.