Acta Materialia 166 (2019) 335–345

Contents lists available at ScienceDirect

Acta Materialia

journal homepage: www.elsevier.com/locate/actamat

Full length article

Establishing structure-property localization linkages for elastic deformation of three-dimensional high contrast composites using deep learning approaches

Zijiang Yang [a], Yuksel C. Yabansu [b], Dipendra Jha [a], Wei-keng Liao [a], Alok N. Choudhary [a], Surya R. Kalidindi [b, c], Ankit Agrawal [a, *]

[a] *Department of Electrical Engineering and Computer Science, Northwestern University, Evanston, IL, 60208, USA*
[b] *George W. Woodruff School of Mechanical Engineering, Georgia Institute of Technology, Atlanta, GA, 30332, USA*
[c] *School of Computational Science and Engineering, Georgia Institute of Technology, Atlanta, GA, 30332, USA*

ARTICLE INFO

Article history:
Received 24 July 2018
Received in revised form
19 December 2018
Accepted 22 December 2018
Available online 31 December 2018

Keywords:
Materials informatics
Convolutional neural networks
Deep learning
Localization
Structure-property linkages

ABSTRACT

Data-driven methods are attracting growing attention in the field of materials science. In particular, it is now becoming clear that machine learning approaches offer a unique avenue for successfully mining practically useful process-structure-property (PSP) linkages from a variety of materials data. Most previous efforts in this direction have relied on feature design (i.e., the identification of the salient features of the material microstructure to be included in the PSP linkages). However due to the rich complexity of features in most heterogeneous materials systems, it has been difficult to identify a set of consistent features that are transferable from one material system to another. With flexible architecture and remarkable learning capability, the emergent deep learning approaches offer a new path forward that circumvents the feature design step. In this work, we demonstrate the implementation of a deep learning feature-engineering-free approach to the prediction of the microscale elastic strain field in a given three-dimensional voxel-based microstructure of a high-contrast two-phase composite. The results show that deep learning approaches can implicitly learn salient information about local neighborhood details, and significantly outperform state-of-the-art methods.

1. Introduction

Material systems used in advanced technologies often exhibit rich heterogeneity over a hierarchy of well-separated length scales. Most material properties are influenced strongly by certain, not yet clearly identified, details of this heterogeneity in the material structure. However, the natural hierarchy of the well-separated material structure length scales allows the adoption of hierarchical multiscale modeling approaches [1–4]. Central to these approaches is the efficient communication of the salient information between the hierarchical structure scales with a sharp focus on the details that strongly influence the overall properties and performance characteristics of the material.

Homogenization deals with the transfer of salient information from a lower material structure scale to a higher material structure scale. Most often this involves capturing the effective (i.e., homogenized) properties defined at the higher material structure scale, while accounting for the heterogeneity that exists at the lower material structure scale. A large number of strategies have been developed and established in literature for addressing homogenization, ranging from the elementary bounding theories [5–7] to the self-consistent approaches [8–10] to the sophisticated statistical continuum theories [11–13].

Localization, on the other hand, deals with the transfer of salient information from a higher material structure scale to a lower material structure scale. As a specific example, it might deal with the microscale spatial distribution of a macroscopically imposed stress (or strain rate) tensor. An illustration of localization in hierarchical multiscale modeling is shown in Fig. 1. As such, localization problems are much more difficult compared to homogenization, and have received only limited attention in current literature [14–23]. Indeed, localization plays an important role in the assessment or prediction of failure-related properties. For instance, both high- and low-cycle fatigue properties of metals are strongly affected by

* Corresponding author.
E-mail address: ankitag@eecs.northwestern.edu (A. Agrawal).

https://doi.org/10.1016/j.actamat.2018.12.045
1359-6454/

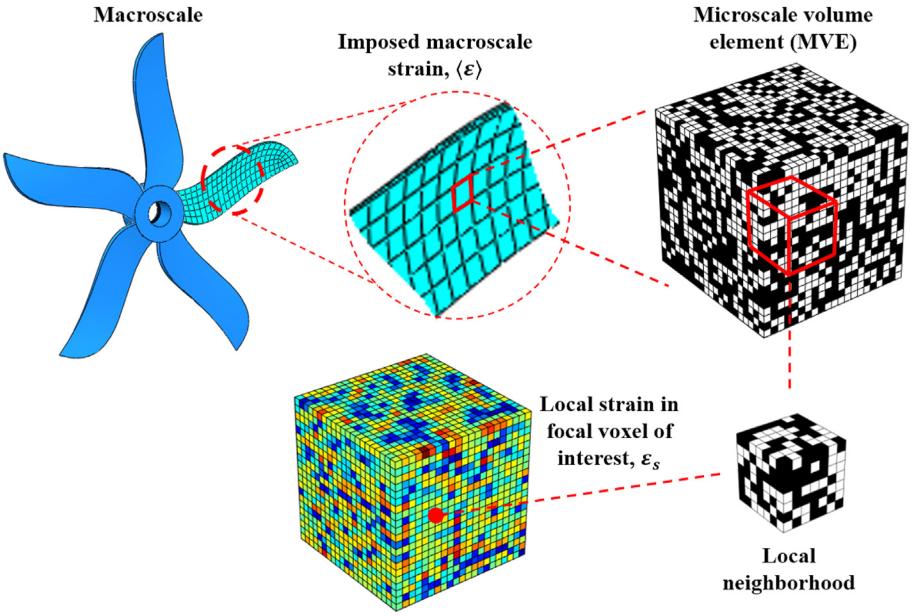
Macroscale
Imposed macroscale strain, $\langle\varepsilon\rangle$
Microscale volume element (MVE)
Local strain in focal voxel of interest, ε_s
Local neighborhood

Fig. 1. Illustration of localization in hierarchical multiscale modeling.

the localization of the stress and plastic strain at the microscale [24–26]. Since the underlying physics behind both homogenization and localization problems is essentially described by exactly the same set of governing field equations, it stands to reason that the localization solutions often automatically embed homogenization solutions within them [27–29].

Localization problems have largely been addressed in literature using either numerical approaches (e.g., the finite element method (FEM) [30,31]) or iterative methods employing Green's functions and fast Fourier transforms (FFTs) [14,16,32]. Neither of these approaches are particularly suited to fast exploration of the vast microstructure design space because of their relatively high computational cost. One of the main drawbacks of these approaches is that they do not focus on learning transferable knowledge from one microstructure to another microstructure. In other words, they are not formulated to take advantage of previously aggregated information on different microstructures to make predictions for new microstructures. One of the most promising approaches to the localization problem is the Materials Knowledge Systems (MKS) approach [17–20,22–24,33] that employs calibrated Green's function based kernels in a non-iterative series solution. The main challenge comes from the difficulty in engineering the local features needed to establish highly accurate localization relationships. In general, the features of interest in establishing localization relationships are all related to the local neighborhood details at the focal voxel of interest (i.e., the voxel under examination) in a given microstructure. In high-contrast composites, the local interactions between microscale voxels extend to larger lengths and demand the consideration of larger neighborhood regions. It was shown in prior work [23] that the number of features that need to be defined for the larger microstructure neighborhood explodes exponentially, posing significant hurdles in the calibration of the kernels through standard machine learning approaches.

In recent decades, deep learning approaches have proven their superior performance over traditional machine learning approaches [34–36]. Deep learning techniques have also recently gained attention in the materials science field, and have been successfully employed in materials image segmentation [37,38], materials data prediction [39–41], homogenization structure-property linkages [42–45], microstructure reconstructions [46,47], and microstructure generation [48,49] with generative adversarial networks (GAN) [50]. In contrast to traditional machine learning approaches, deep learning offers an end-to-end framework where it usually takes raw data as input (i.e., without the need for any feature engineering). Because of its flexible structure and remarkable learning capability, deep learning can automatically extract higher-order information embedded in the raw input. It was shown in prior work [44] that convolutional neural networks (CNN) can automatically extract important higher-order microstructure statistics (i.e., up to 1000-point statistics) central to establishing reliable homogenization structure-property linkages for high contrast material systems. The extraction of this level of information is not computationally feasible through standard feature extraction methods. In this work, we focus on two high-contrast composite material systems, with contrast values of 10 and 50 (these represent the ratios of Young's moduli of the constituent phases in the composite). More specifically, we propose a deep learning model to efficiently and accurately predict microscale elastic strain field of three-dimensional (3-D) high contrast elastic composites. In order to critically evaluate the performance of proposed model, the results obtained via deep learning approach are compared to three different benchmark approaches that solve the localization problem via other data-driven techniques [19,20,22].

2. Benchmark methods for localization

Elastic localization has been modeled for 3-D microstructures through single and multi-agent feature extraction methods [19,20], and the materials knowledge system (MKS) framework [22]. These methods are used in this study as benchmarks to evaluate the performance of deep learning methods, and are briefly described next.

2.1. Feature extraction methods

The first two benchmarks used in this paper are based on feature extraction methods [19,20]. Liu et al. [19] extracted a relatively large set of features to identify the local neighborhood around the focal voxel of interest, applied feature ranking methods to establish a reduced set of features that preserve the most significant information, and then used random forest techniques to predict microscale elastic strain field. However, such a single-agent feature extraction method does not effectively capture the embedded correlations in the trained data. Thus, in a following study [20], Liu et al. developed a multi-agent feature extraction method to further improve the model's performance. More specifically, this method consisted of two steps. In the first step, the ensemble of microstructures were divided into categories using metrics such as volume fraction and pair-correlation functions so that these different categories would have low inter-similarity and high intra-similarity. In the second step, the reduced set of features in Ref. [19] were used to characterize the microstructures in each category and the random forest algorithms were extended separately to these categories to predict microscale elastic strain field. The multi-agent feature extraction approach did result in a trained model that exhibited higher accuracy compared to the single-agent feature extraction methods.

2.2. First-order localization series of MKS framework

Another data-driven method used as benchmark in this paper is the localization series of MKS framework. MKS framework addresses the localization problem in the form of a series expansion derived from the statistical continuum theories [12,13]. Each term in the series is defined as a convolution product of the Green's function based kernel over the microstructure descriptor. For the elastic localization problem, the first-order MKS expansion for a macroscale imposed strain component $\langle \varepsilon \rangle$ can be written mathematically as

$$\varepsilon_s = \left(\sum_r \sum_h \alpha_r^h m_{s+r}^h \right) \left\langle \varepsilon \right\rangle \tag{1}$$

where m_s^h provides the digital representation of the microstructure (reflects the volume fraction of material local state h occupying the spatial bin or voxel s [51]), r systematically indexes all of the neighboring voxels near the focal voxel of interest, α_r^h denote the model fit parameters (or weights) that need to be trained using the available data, and ε_s denotes the localized strain (model output) in the focal voxel of interest. In the case where each spatial voxel is occupied by one distinct phase, m_s^h takes the value of either 0 or 1 [22,52–55]. In the MKS framework, α_r^h are called influence coefficients and are usually calibrated to the data aggregated from executing finite element simulations on a large number of digitally created microstructures. The model calibration is often pursued using standard linear regression techniques. There are two major distinctive features of the model form presented in Eq. (1): 1) The influence coefficients are defined only in terms of relative distance r to the focal voxel of interest, s. Hence, they are independent of microstructure morphology. Once they are obtained, they can be used to predict the response field of any new microstructure of the same material system. 2) Since the expression shown in Eq. (1) involves a convolution product, fast Fourier transform (FFT) algorithms can be exploited for computational efficiency. Consequently, the MKS predictions for new microstructures can be obtained several orders of magnitude faster compared to the physics-based finite element approaches.

3. Convolutional neural networks for localization

Convolutional neural networks (CNN), introduced by LeCun et al. [56], are widely applied in computer vision field to solve various difficult problems related to feature recognition and image segmentation [57–62]. A conventional CNN model usually has a stack of several convolutional layers and pooling layers. The convolutional layers employ filters to learn the important local features of images along with their relative importance. The pooling layers reduce dimensionality, while preserving the most important information. The output of multiple stacks of convolutional and pooling layers are then fed into fully connected layers which have the final list of predictors on which the model is trained. In this study, CNNs are built for the elastic localization in 3-D microstructures of high contrast composite microstructures using the typical CNN architecture illustrated in Fig. 2.

3.1. Convolutional layer

Convolutional layer is the core component of CNN and it uses filters to extract the salient features from images. The readily accessible implementations of CNN employ 2-D filters on 2-D images. Since our application involves 3-D microstructures, we need to find a suitable workaround for this limitation that does not significantly increase the computational cost involved in the training of the model. In this study, we accomplished this by treating the 3-D neighborhood around the focal voxel as multiple channels of 2-D layers, where each layer is then convolved with a 2-D CNN filter. While this approach provides us a pseudo 3-D filter, it should be recognized that it is not exactly the same as using 3-D filters. For example, the trained model in the approach used here is likely to be sensitive to which planes are selected for converting the 3-D neighborhoods into stacks of 2-D layers. In this work, the 2-D layers are selected perpendicular to the maximum principal strain direction.

A general convolution layer in the present implementation takes as input N channels of 2-D images, where each image (i.e., each channel) is defined on a uniform square grid of voxels. The N channels of 2-D images are then convolved with multiple sets of N 2-D CNN filters, each of size $m \times m$. The common practice is to use an odd value for m (ensures symmetry in the number of voxels on either side of the central voxel in the convolution). The convolution employed in this process can be mathematically expressed as

$$O(x,y) = \sum_{k=0}^{N-1} \sum_{j=0}^{m-1} \sum_{i=0}^{m-1} I\left(x - \frac{m-1}{2} + i, y - \frac{m-1}{2} + j, k \right) \times w(i,j,k) + b \tag{2}$$

where I and O denote the input and output, x and y index the position of voxels in an input 2-D image, and w and b denote the corresponding weights and bias captured by a set of filters (to be trained). It should be noted that each set of 2-D filters produces a 2-D output image (by using Eq. (2) for all x and y values in an input 2-D image). Each 2-D output image from the convolution then serves as an independent input channel for the next convolution layer.

In CNN approaches, the output O from Eq. (2) is passed through an activation function to obtain the final output at voxel (x,y) from the convolution layer (i.e., the use of an activation function is an inherent component of the convolution layer). The main purpose of the activation function is to capture the nonlinear complex functional mappings between input and output (note that the convolution operation described above involved only a linear mapping). One of the most commonly used activation function in deep learning is ReLU (rectified linear unit) [63] formulated as $f(x) = max(0,x)$. This activation function is applied to all voxels in the 2-D output image from the convolution, and the output image is then

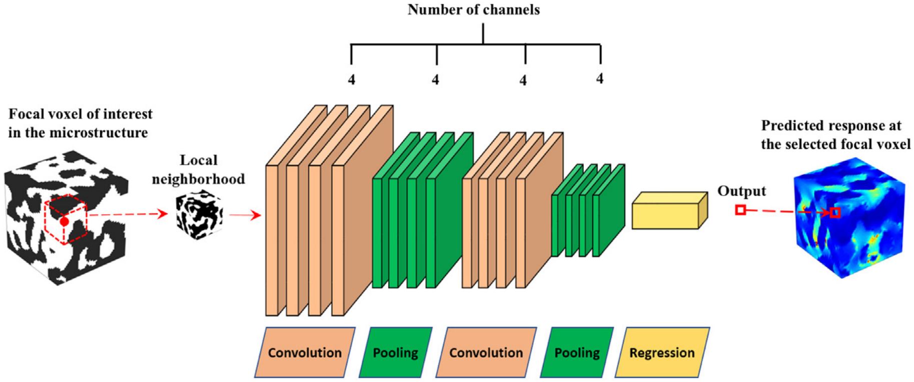

Fig. 2. Illustration of a typical convolutional neural network (CNN) used in this study for building localization linkages.

called a feature map. Thus multiple feature maps (one per channel per convolutional layer) produced in this manner are expected to capture the diverse range of salient features in the input that are strongly correlated to the selected output. Note that if one desires to keep the size of the input images (describing the neighborhood of a focal voxel) and of the output feature maps of the convolutional layer as same, zero padding needs to be applied when considering focal voxels in the boundary of the input microstructure. The alternative to zero-padding is to allow the shrinking of the output feature maps by ignoring focal voxels in the boundary regions of the given representative volume element of the microstructure.

3.2. Pooling layer

Pooling layer is used to reduce the dimensionality, while retaining the most important information in the data. Average pooling is one of the most widely used pooling operations, and essentially amounts to coarsening of the image. Mathematically, one can express this as

$$O(x,y) = \sum_{j=0}^{s-1} \sum_{i=0}^{s-1} \frac{I(s\times x+i, s\times y+j)}{s^2} \tag{3}$$

where s denotes the coarsening length scale (in number of voxels), and (x,y) identifies a voxel in the output of the feature map after applying the average pooling layer. The pooling layer is applied for each feature map, which means the number of input and output feature maps is the same. In most cases, the pooling layer results in a reduction of the number of the spatial voxels by an integer factor. The most important advantage of the pooling operation is that it reduces the computational requirements in training the mathematical model (i.e. regression, classification, etc.) as it reduces the number of dimensions flowing from convolutional layers to the fully connected layers. This also helps in the case of regression problems where overfitting is a major problem, especially for cases involving a large number of features.

3.3. Fully connected layer

In the final stage of CNN, the output of several stacked convolutional layers and/or pooling layers are flattened to a one dimensional vector, and this vector is fed into a fully connected layer. A fully connected layer is the same as a layer of conventional artificial neural network, which consists of multiple neurons. The operation of a neuron can be formulated as

$$o = f\left(\sum_{i=0}^{n-1} x(i)\times w(i)+b\right) \tag{4}$$

where o represents the output of this neuron, $x(i)$ denotes the i^{th} input, n is the number of inputs to the fully connected layer, and $f(.)$ represents the activation function. As in convolutional layer, ReLU is the most commonly used activation function in the fully connected layer, except that in the final output layer, a linear activation function is used for regression problems and softmax activation function is used for classification problems.

3.4. Proposed deep learning model

It is logical that the details of the local neighborhood strongly influence the accuracy of predictions from localization models. This neighborhood information is usually referred as the higher-order microstructure information [51,64]. It is also well-known that more higher-order information (i.e., more details of the neighborhood) are needed for localization models in high contrast composite material systems. In Refs. [19,20], a 5 × 5 × 5 cube centered on the focal voxel was used to build the localization models for a composite with a contrast of 10 in the Young's moduli of its constituents. In order to predict microscale elastic strain field of higher contrast composites studied here, a 11 × 11 × 11 cube centered on the focal voxel was taken as input to the deep learning model. As already mentioned, this means that the input is treated as 11 channels of 2-D images of size 11 × 11 × 11 voxels. The input images had only binary values, i.e., 0s and 1s. The input images were transformed such that the values at each voxel were assigned as −0.5 or 0.5 (replacing 0 and 1, respectively) as this was found to improve the performance of the convolution layers. In addition, the local strains were scaled by a factor of 10000 to avoid numerical issues associated with their small magnitudes (recall that these are elastic strains).

Different CNNs and CNN architectures were explored to find the best trained model for the present application. The architecture of deep learning model is crucial to determine its learning capability. Thus different architectures of CNNs consisting of convolutional layers with zero-padding, pooling layer and fully connected layers, were first explored. More specifically, the depth of CNN was gradually increased from 7 layers to 18 layers by stacking additional convolutional layers and pooling layers. However, the accuracy of these trained models was inadequate, and did not improve much with these changes. There might be two reasons for this relatively poor performance: (1) The size of input image is relatively small compared with images in conventional computer vision field.

Consequently, reducing the image size by using pooling layer might lose too much microstructural information. (2) Due to fact that input images are binary and small, all the voxels in the images could have significant effect on the final predictions. Thus adding 0s around the boundary by using zero-padding might confuse the training process. Therefore, we decided to use a six-layer CNN that consisted of convolutional layers without zero-padding and fully connected layers. Because convolutional layer is the core component of CNN and it uses filters to capture the salient features from the images, the number of filters in each convolutional layer was systematically varied. More specifically, the number of filters in each convolutional layer was gradually increased to find the best number of filters in each convolutional layer for the present application. After that, the number of neurons in fully connected layers was systematically varied so that the salient features captured by the conovlutional layers could be effectively utilized by fully connected layers to make accurate predictions. Finally, $L2$ regularization with different penalty strengths were explored to avoid overfitting.

The architecture of the best CNN model resulting from the many trials conducted for the present application is shown in Table 1. In this model, convolutional layers 1 and 2 have 128 and 256 3×3 filters, respectively, and zero-padding is not employed in the convolution operation. The convolutional layers are followed by two fully connected layers where the first and second layers contain 2048 and 1024 neurons, respectively. All the weights in the CNN are initialized by normalized initialization [65]. This initialization method samples a $U[-r, r]$ with $r = \sqrt{\frac{6}{fan\_in + fan\_out}}$ where fan_in and fan_out are the number of inputs and outputs of the layer, respectively. ReLU is used as activation function for all the convolutional layers and fully connected layers except the output layer where linear activation function is used. In order to avoid overfitting, $L2$ regularization with 0.0001 penalty strength is applied in each convolutional layer and fully connected layer.

During the training, batch size was set as 252 data points (i.e., 252 $11 \times 11 \times 11$ cubes centered at corresponding focal voxels were propagated through the network in each training iteration), and mean squared error was used as loss function. Adam optimizer [66] was used as it is a more effective optimization algorithm than conventional stochastic gradient descent algorithm. For the parameter settings of Adam, the learning rate was 0.001, β_1 value was 0.9 and β_2 value was 0.999. Early stopping technique was applied to determine when to terminate the training process. More specifically, the training process was terminated when the value of loss function on validation set did not improve over 10 epochs (i.e., 10 complete passes through the entire dataset).

There is a post-processing step after the training is terminated and the predictions are computed. Since the total strain on a MVE of FE model is fixed (i.e., 0.001 for both contrast 10 and 50 composites materials datasets discussed next), the predicted local strain of focal voxel is multiplied by a scaling factor $0.001/e$ where e is the total predicted strain on the MVE so that the total strain on each MVE of ground truth (i.e., FE model) and predictions are the same (i.e., the total predicted strain on each MVE is 0.001). The datasets and the performance of the proposed deep learning models are presented in the next section.

Table 1
The dimensionality of each layers in the proposed CNN (bs. Denotes the batch size).

Layer	Dimension
Input Layer	$bs. \times 11 \times 11 \times 11$
Convolutional Layer 1	$bs. \times 9 \times 9 \times 128$
Convolutional Layer 2	$bs. \times 7 \times 7 \times 256$
Fully Connected Layer 1	$bs. \times 2048$
Fully Connected Layer 2	$bs. \times 1024$
Output Layer	$bs. \times 1$

4. Results and discussion

4.1. Datasets

As mentioned before, the goal of this paper is to predict microscale elastic strain field for 3-D high contrast elastic composites using CNN approaches. In this study, we used digitally generated microstructures and finite element (FE) tools to produce the data needed to train the linkages. Even though using 3-D microstructures obtained via experimental methods sounds more attractive, such datasets are scarce. Since deep learning approaches require more datasets than traditional machine learning techniques, accumulating a training set from experiments is expensive and time-consuming. Hence, simulation datasets are employed in this study to explore the viability of the deep learning techniques for predicting microscale elastic strain field for 3-D microstructures.

To train CNN for localization, two ensembles of microstructures with contrasts of 10 and 50 were generated. Each 3-D volume of microstructure is referred to as microscale volume element (MVE) and they were discretized into a uniform grid of size $21 \times 21 \times 21$. Each voxel is occupied by one of the phases. The soft and hard phase are indicated by 0 and 1, respectively. The phases were assigned to the voxels in a randomized manner. The contrast 10 ensemble have MVEs with volume fractions ranging from 0.01% to 99.43% for hard phase. On the other hand, the 50 contrast ensemble had volume fractions in the range of 0.01%–99.99% for hard phase.

Once the MVEs were generated, finite element (FE) simulations were executed by using the commercial software of ABAQUS [67]. Each voxel in the MVE was converted into a 3-D 8-noded C3D8 element [67]. All simulations employed periodic boundary conditions where all macroscale strain values had zero values except the normal strain in direction 1 (i.e., $\langle \varepsilon_{ij} \rangle = 0$ except $\langle \varepsilon_{11} \rangle \neq 0$ where $i, j = 1, 2, 3$ and $\langle \rangle$ represent the volume average). The macroscale normal strain component in direction 1 is taken as $\langle \varepsilon_{11} \rangle = 0.001$ for both ensembles. The local response field of interest was selected as ε_{11}. It should be noted that the strategy proposed in this paper can be repeated for a total of six macroscale strain conditions to obtain the full field of deformation using the superposition principle [68,69].

As mentioned before, two ensembles of datasets with contrasts of 10 and 50 were generated to evaluate the viability of deep learning approaches developed for elastic localization linkages. Both microscale constituents (i.e., distinct phases) were assumed to be isotropically elastic and the Poisson ratio, ν, for both phases in both ensembles was taken as 0.3. The contrast in elastic composites was defined as the ratio between the Young's modulus of the two distinct phases. For contrast-10 dataset, the Young's modulus of hard and soft phases were selected as $E_1 = 120GPa$ and $E_2 = 12GPa$, respectively. On the other hand, for contrast-50 dataset, the modulus of the hard phase is kept the same while the soft phase was assigned a Young's modulus of $2.4GPa$. An example MVE with contrast-10 and its elastic strain field of ε_{11} is depicted in Fig. 3.

For the ensemble of contrast-10, 2500 MVEs were generated. 1200 and 300 of these MVEs were used for training and validation of CNN, respectively. The remaining of 1000 MVEs were not included in the training and used only as a testing set. To be more specific, there were 100 vol fraction categories and 25 MVEs in each volume fraction category for this dataset. For each volume fraction, 12 MVEs were randomly selected for training set, 3 MVEs were randomly selected for validation set and rest 10 MVEs were used for testing set. In this way, all the three sets contained representations from all the volume fraction. Each MVE has $21 \times 21 \times 21 (= 9261)$

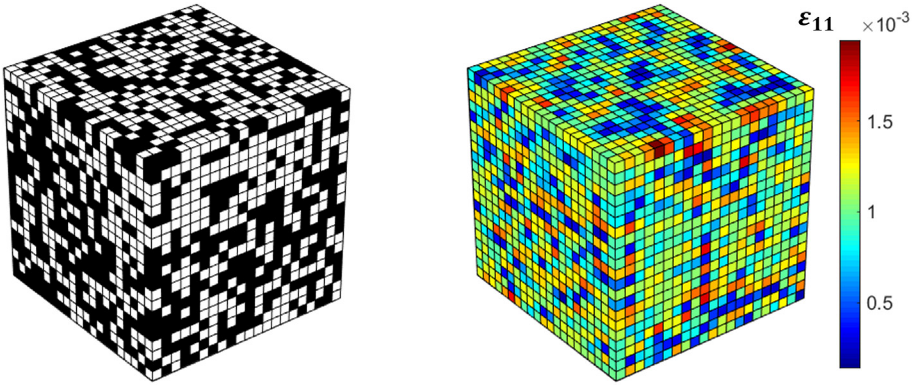

Fig. 3. Visualization of an example contrast-10 MVE (left) and its strain field (right). The white and black voxels in the MVE correspond to hard and soft phases, respectively.

voxels where each voxel provides one data point for building and evaluating the desired models. Hence, training, validation and test sets had $1200 \times 9261(= 11,113,200)$, $300 \times 9261(= 2,778,300)$, and $1000 \times 9261(= 9,261,000)$ voxels, respectively. On the other hand, 3000 MVEs were generated for contrast-50. Using the similar data splitting strategy, this ensemble was split into 2000 training, 500 validation and 500 testing MVEs, respectively. This resulted in $2000 \times 9261(= 18,522,000)$, $300 \times 9261(= 2,778,300)$, and $500 \times 9261(= 4,630,500)$ data points for training, validation, and testing of the ensemble of contrast-50. Both datasets are large enough to develop deep learning models.

We used Python 2.7 and Keras [70], which is a high-level neural networks library built on top of TensorFlow [71] to implement deep learning model. Scikit-learn [72] and PyMKS [73] are used for implementing benchmark methods. CNN trials were carried out on a NVIDIA DIGITS DevBox with 4 TITAN X GPUs with 12 GB memory for each GPU and Core i7-5930 K 6 Core 3.5 GHz CPU with 64 GB DDR4 RAM. In order to get a fair comparison, model performance is evaluated by the mean absolute strain error (MASE) [19], which is defined as below,

$$e = \frac{1}{S}\sum_{s=1}^{S}\left|\frac{p_s - \widehat{p}_s}{\langle \varepsilon_{11} \rangle}\right| \times 100\% \tag{5}$$

where $\langle \varepsilon_{11} \rangle$ is the macroscopic strain tensor component applied to MVE via periodic boundary conditions. This metric is used to measure average error for a single MVE. p_s and $\widehat{p}_s$ denote the local strain in the voxel s from the FE model and CNN model, respectively. The performance of predicting microscale elastic strain field associated with CNN is also compared to benchmark methods using the same error metric.

4.2. Results of contrast-10 composites dataset

Table 2 shows the comparison of the average MASE of entire testing set for the proposed deep learning model and the benchmark methods. The proposed deep learning model achieves 3.07% average MASE for the entire testing set, and the standard deviation of MASE across all the MVEs in the testing set is 1.22%. By comparing with benchmark methods, we observe that the proposed deep learning model improves the previous best prediction performance in terms of MASE by as much as around 61.8% (1−3.07/8.04), which shows that deep learning model can produce a more accurate predictive model. Further, to evaluate and quantify the robustness of the proposed deep learning model, we constructed and evaluated 10 deep learning models on different training-testing splits (but same hyper-parameters). The average MASE across the 10 trials is 3.23% with a standard deviation of 0.22%, which is relatively small, suggesting that the proposed deep learning model is quite robust. In addition, in Ref. [20], training data is divided into several groups by some dividing criterion and then individual model is developed for each data group. However, the proposed deep learning model is directly trained on overall training set, which produces a model with better generalization.

Table 2
Results comparison of different models for contrast-10 dataset (standard deviations are available only for MKS method and deep learning model because studies describing other benchmarks did not present results of standard deviation).

Method	Average MASE for testing set
Single-agent based method [19]	13.02%
MKS method	10.86%±4.30%
Multi-agent based method [20]	8.04%
Deep learning model	3.07%±1.22%

Fig. 4 shows the plot of MASE for each individual MVE in testing set versus volume fraction with error bars of contrast-10 composites dataset using the proposed deep learning model. We can observe that the MASE for MVEs with around 40% volume fraction for hard phase has the largest error and variance (i.e. large average MASE and standard deviation).

The results in Fig. 4 assess the performance of deep learning model (DL) on an average basis. Since the problem at hand is elastic localization, we also compare the local strain fields obtained from FE method to DL and MKS methods. The comparison of FE, DL and MKS methods at the level of individual voxel is shown in Fig. 5 for two slices selected to represent results from the best and worst predictions (based on MASE values of DL). For the MVE with best MASE, the strain field histogram of DL predictions matches that of ground truth very well, except that the curve of Phase 1 prediction shifts slightly to the right. The difference of the strain map of a randomly selected slide between DL predictions and ground truth is indistinguishable, while the performance of MKS method is significant worse. On the other hand, even for the MVE with worst MASE, the trend of strain field histogram of DL predictions matches that of ground truth very well. The differences of DL prediction and ground truth in the strain field maps are unnoticeable, while MKS method tends to underestimate the strain field.

4.3. Results of contrast-50 composite dataset

The same architecture and hyperparameters used to train the ensemble of contrast-10 were also used to train the CNN on the ensemble of contrast-50. For this case study, it was only possible to

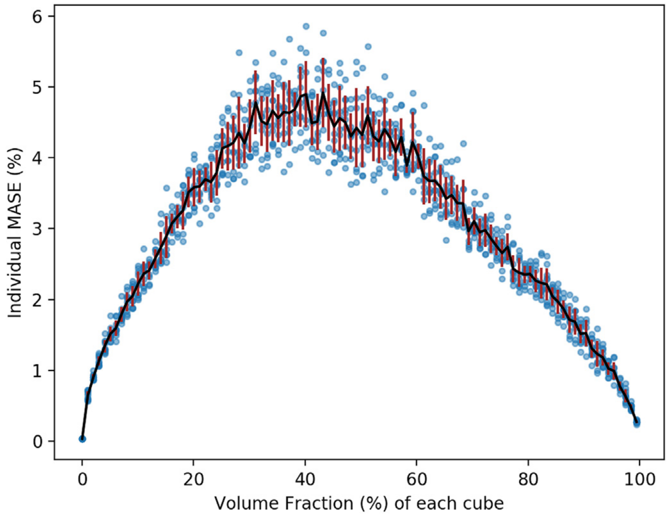

Fig. 4. Plot of MASE for each individual MVE in testing set versus volume fraction with error bars of contrast-10 dataset using the proposed deep learning model.

use one benchmark as the studies describing other benchmarks did not present results on contrast-50 dataset, nor is their code available. The results in Table 3 show the comparison of average MASE of entire testing set between proposed deep learning model and the first-order MKS method. The proposed deep learning model achieves 5.71% average MASE on entire testing set, and the standard deviation of MASE across all the MVEs in the testing set is 2.46%. In contrast, the MASE of the benchmark method is 26.46%, which is significantly higher. In other words, the proposed deep learning model improves the performance by as much as around 78.4% (1−5.71/26.46) in terms of MASE. Further, to evaluate and quantify the robustness of the proposed deep learning model, we constructed and evaluated 10 deep learning models on different training-testing splits (but same hyper-parameters). The average MASE across the 10 trials is 5.38% with a standard deviation of 0.22%, which is again relatively small, suggesting that the proposed deep learning model is quite robust for contrast-50 dataset as well.

Table 3
Results comparison of different models for contrast-50 dataset.

Method	Average MASE for testing set
MKS method	26.46%±6.91%
Deep learning model	5.71%±2.46%

The superior performance of the CNN approach over the MKS method indicates tremendous promise for a feature-engineering free approach to predict microscale elastic strain field. This is of tremendous value to multiscale materials design efforts in higher contrast composites as the identification of the salient features has been the central hurdle. Furthermore, the fact that the same CNN architecture has provided excellent predictive models for both contrast-10 and contrast-50 datasets supports the conclusion that the use of deep learning approaches offers a higher generalization.

Fig. 6 shows the plot of MASE for each individual MVE in testing set versus volume fraction with error bars of contrast-50 composites dataset using the proposed deep learning model. The MASE for MVEs with around 23% volume fraction for hard phase has the largest error and variance (i.e. large average MASE and standard deviation), while the MVEs with around 80% volume fraction for hard phase has the best predictions (i.e. small average MASE and standard deviation).

To visually demonstrate the predictive power of deep learning model (DL) for contrast-50 dataset, the MVEs with best MASE and worst MASE are selected. Fig. 7 shows the comparison of FE, DL and MKS method at the level of individual voxels. The comparisons are

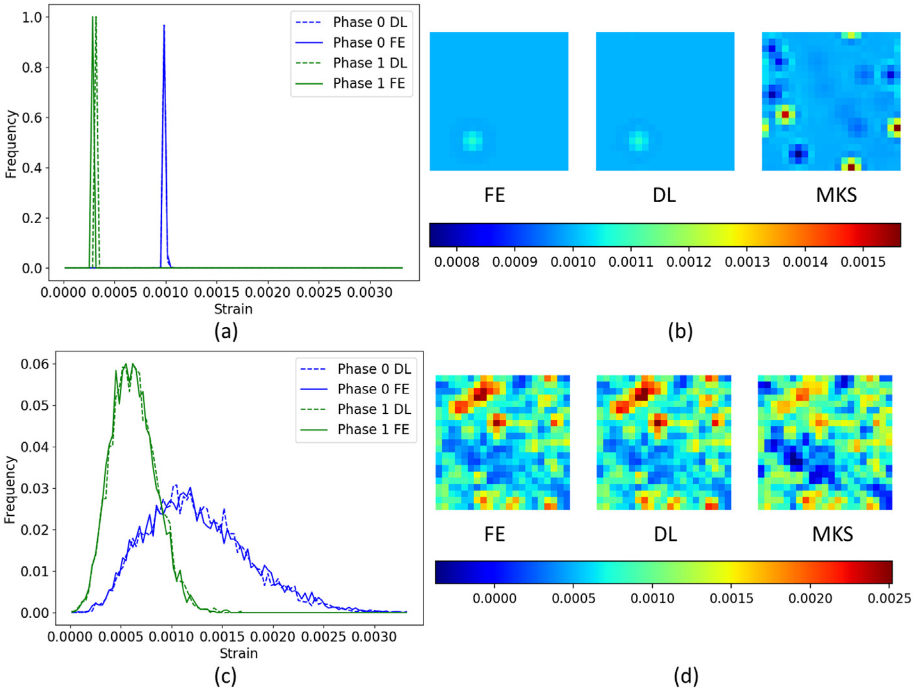

Fig. 5. Comparison of FE model (i.e. ground truth), deep learning (DL) method and MKS method predictions for contrast-10 dataset. (a) Strain fields histogram of the MVE with the best MASE for FE and DL. (b) Strain field map of a randomly selected slide in the MVE with the best MASE for FE, DL and MKS. (c) Strain fields histogram of the MVE with the worst MASE for FE and DL. (d) Strain field map of a randomly selected slide in the MVE with the worst MASE for FE, DL and MKS.

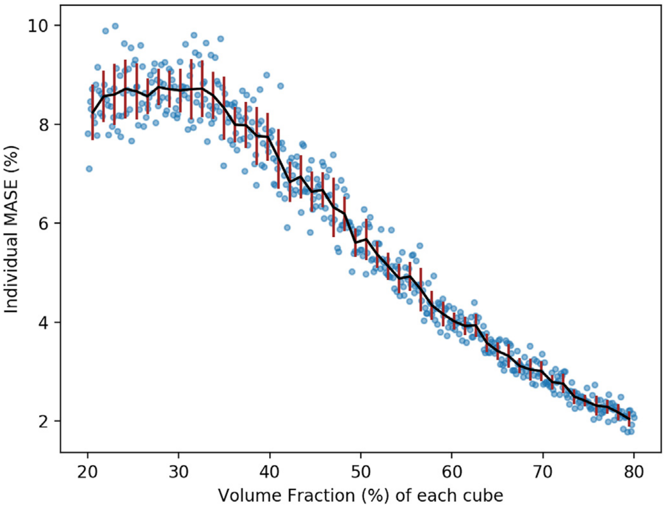

Fig. 6. Plot of MASE for each individual MVE in testing set versus volume fraction with error bars of contrast-50 dataset using the proposed deep learning model.

made for a randomly selected slice from 3-D MVEs that exhibited the best and the worst MASE of DL. For the MVE with the best MASE, the trend and the peak of the strain field histogram of DL predictions and ground truth accurately match each other. The difference between FE and DL is almost indistinguishable from each other, while the MKS method tends to overestimate the strain field. For the MVE with the worst MASE, the overall trend of strain field histogram of DL predictions and ground truth match each other very well. Even the strain maps are quite similar to each other demonstrating the fidelity of the DL models for high contrast composites, while the MKS method underestimates the strain filed resulting in a significant error.

4.4. Interpretation of what the deep learning model learns

Though deep learning model has a striking learning capability that is superior to the traditional machine learning methods, it usually works like a black box due to its complex architecture and millions of parameters. Thus, the interpretation of what the deep learning model actually learns merits attention. In this section, we adjust the inputs of the model and interpret what the deep learning model has learned by analyzing their corresponding predictions.

In Ref. [19], Liu et al. identify groups of neighboring voxels based on their distance from the focal voxel of interest. In this classification scheme, L level neighbors include all the voxels that are at a center-to-center Euclidean distance of square root of L from the voxel of interest (treating each cubic voxel to be of unit length). Fig. 8 illustrates examples of the first three level neighbors. Empirically, lower level neighbors have larger effect on the response of the focal voxel. In order to see if the proposed deep learning model can capture this information, we adjust the input data by setting the values of different level neighbors as 0. Note that the inputs have been rescaled from $[0,1]$ to $[-0.5,0.5]$ during the training, thus setting the value as 0 eliminates the contribution from that voxel. Then we can predict the modified response, and assess the percentage contribution from the different levels of neighbors. For this purpose, we define a residual percentage E

$$E=\frac{|\widehat{y}-y|}{y}\times 100\% \tag{6}$$

where $\widehat{y}$ and y are prediction and ground truth, respectively. A total of 11 MVEs from the contrast-10 testing set were selected with a difference of volume fraction around 10% (i.e., their volume fractions are around 1%, 9%, 19%, 29%, …,99%). Similarly, 11 MVEs from the contrast-50 testing set were also selected with a difference of volume fraction around 6% (i.e., their volume fractions are around 20%, 26%, 32%, 38%, …, 80%). For each voxel in all these MVEs, we

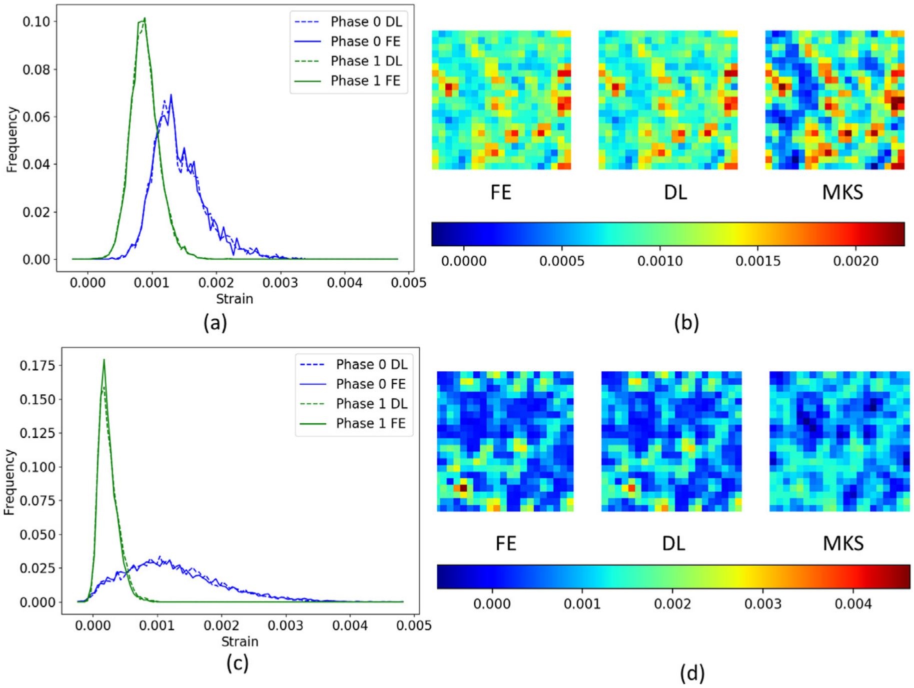

Fig. 7. Comparison of FE model (i.e. ground truth), deep learning (DL) method and MKS method predictions for contrast-50 dataset. (a) Strain histograms of the MVE with the best MASE for FE and DL. (b) Strain field map of a randomly selected slide in the MVE with the best MASE for FE, DL and MKS. (c) Strain histograms of the MVE with the worst MASE for FE and DL. (d) Strain field map of a randomly selected slide in the MVE with the worst MASE for FE, DL and MKS.

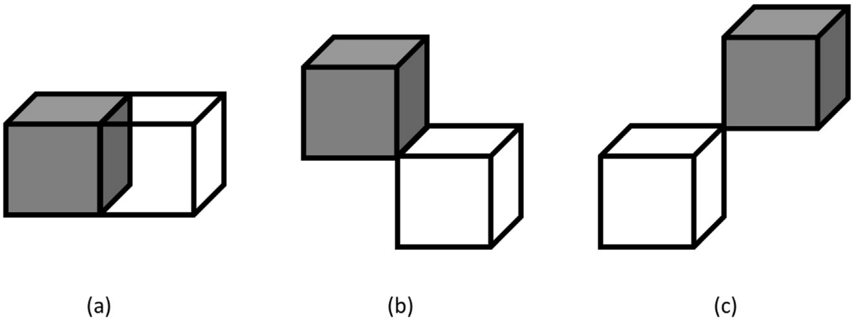

Fig. 8. Illustration of different level neighbors. (a) first, (b) second, and (c) third level neighbor.

compute the residual percentage *E* of original data input, then repeated the computations by removing first, second, third, and up to twenty first neighbor level, one at a time.

Fig. 9 shows the plots of residual percentage after removing different level neighbors for both contrast-10 and contrast-50 datasets. The plots show that the residual percentage generally decreases with increasing level of neighbors, and the first and second level neighbors have the most significant influence on the response of focal voxel of interest. Since the CNN was not explicitly provided this knowledge in any manner, it is indeed remarkable that the CNN model automatically reflects this knowledge. Note that the CNN model is able to capture this knowledge at both contrast levels. It is also remarkable that the CNN model captured higher levels of interactions at a given *L* in the higher contrast composite compared to the lower contrast composite. Once again, the CNN model learned this knowledge also by itself. In addition, the curves in the plots are not decreasing monotonically. This means that the distance from focal voxel of interest is not the only factor that decides the significance of neighbors. In other words, the contributions from the higher level neighbors could be more significant than that from lower level neighbors due to their different directional relation to the focal voxel of interest. The trained CNN model is thus found to have implicitly learned all of this salient information to make accurate predictions.

The CNN models presented in this work show their superiority compared to traditional machine learning approaches mainly in two aspects. 1) Accuracy: the results show that the deep learning approaches can improve the model's performance by more than 50% compared with traditional machine learning approaches, which attests to their high learning capabilities when there is a sufficiently large dataset. 2) Generalization: in order to get better accuracy, multi-agent learning strategy is usually used in traditional machine learning method. Because the variance of the data is decreased after it is divided into multiple data clusters, such processing could deteriorate the generalization of the model. In addition, complex problem-specific features could also make the model difficult to generalize to other material systems. In contrast, deep learning has the capability to directly train on entire raw dataset without feature engineering and extract enough useful microstructure information, which results in a model with better generalization. As we know, data is usually limited in material science research, which hinders the applications of deep learning in the field. Thus, transfer learning [74] can provide a promising alternative. Because the 3-D microstructures used in this work do not have any special constraints, such as volume fraction, orientation, and shape of either soft or hard phase, the features learned by the proposed model are general enough to well characterize general two-phase microstructures. Thus, the proposed model can serve as a pre-trained model for transfer learning to assist researchers in developing models to predict materials properties of 3-D two-phase microstructures. Therefore, it can accelerate the applications of deep learning in material science research. In addition, considering the advantages mentioned above, the other two main use cases for the proposed data-driven model could be, 1) It can allow rapid evaluation of a large number of microstructures in designing new materials meeting desired mechanical responses. This is because the data-driven models are computationally cheap compared to the numerical tools such as FEM models. In fact,

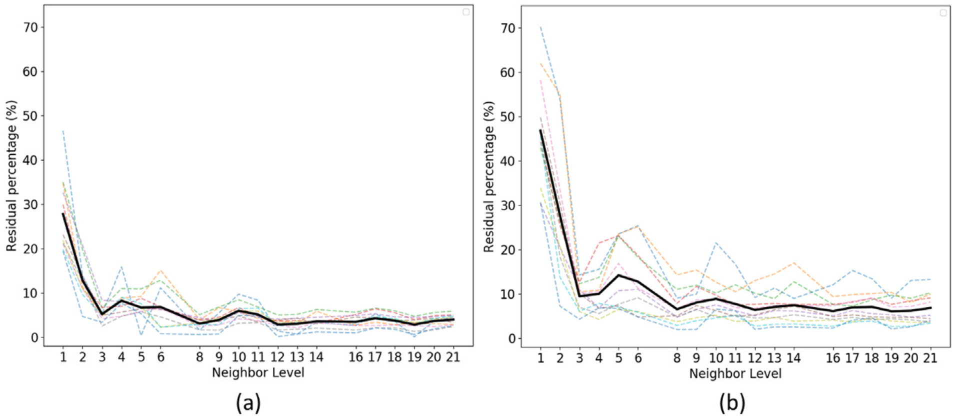

Fig. 9. Plots showing contributions of the different level neighbors in the trained CNN models. The dashed lines in both figures show the average percentage contributions of selected MVE in the datasets, and the solid lines in both figures show the average residual percentage of a collection of selected MVEs in the datasets. (a) plot for contrast-10 dataset. (b) plot for contrast-50 dataset.

design of material microstructure using FEM models is prohibitively expensive and its viability has not been demonstrated in any prior literature. However, the data-driven models have the potential to address this gap [75–77]. 2) The data-driven models can also help in objective calibration of the many parameters present in the physics-based microscale models to the often limited amount of experimental data [78–80]. However, there are also limitations for this work. Because large amount of experimental data is not available, the proposed model is trained only on simulation datasets. The effectiveness of the proposed model still needs to be validated with suitable experimental data. If the proposed model could also be trained and validated with experimental data, it can offer a valuable, low cost, and accurate predictive tool for multiscale material design efforts.

5. Conclusions

In this paper, a convolutional neural network is developed and tested on both contrast-10 and contrast-50 composite datasets. The results show that the proposed deep learning model significantly outperforms benchmark methods by 61.8% and 78.4% in terms of average MASE of testing set on contrast-10 and contrast-50 datasets, respectively. It is clear from these trials that deep learning is a promising technique to build a feature-engineering-free, high accuracy, low computational cost, and high generalization model to study PSP linkages in complex materials systems. Moreover, since the proposed deep learning architecture worked well for a large range of contrasts, it can be used as a pre-trained model with transfer learning approach to establish PSP linkages for other high contrast composites materials when the dataset is limited. There are several possible directions for future works based on this study. As an example, we only show the relative importance of neighbors based on the distance from focal voxel in section 4.4, one could further investigate the relative importance of neighbors based on other criterion, such as the shape of phase, which might provide more insights about relationship between focal voxel and its neighbors. Such insights if established reliably could lead to a new easy way to implement design rules for microstructures (aimed at meeting a desired property).

Acknowledgements

This work is supported in part by the following grants: AFOSR award FA9550-12-1-0458; NIST awards 70NANB14H012 and 70NANB14H191; NSF award CCF-1409601; DOE awards DESC0007456, DESC0014330; and Northwestern Data Science Initiative.

References

[1] S. Groh, E. Marin, M. Horstemeyer, H. Zbib, Multiscale modeling of the plasticity in an aluminum single crystal, Int. J. Plast. 25 (8) (2009) 1456–1473.
[2] V. Vaithyanathan, C. Wolverton, L. Chen, Multiscale modeling of precipitate microstructure evolution, Phys. Rev. Lett. 88 (12) (2002) 125503.
[3] S. Chandra, M. Samal, V. Chavan, S. Raghunathan, Hierarchical multiscale modeling of plasticity in copper: from single crystals to polycrystalline aggregates, Int. J. Plast. 101 (2018) 188–212.
[4] J. Knap, C. Spear, K. Leiter, R. Becker, D. Powell, A computational framework for scale-bridging in multi-scale simulations, Int. J. Numer. Methods Eng. 108 (13) (2016) 1649–1666.
[5] X. Wu, G. Proust, M. Knezevic, S. Kalidindi, Elastic–plastic property closures for hexagonal close-packed polycrystalline metals using first-order bounding theories, Acta Mater. 55 (8) (2007) 2729–2737.
[6] P.P. Castañeda, Second-order homogenization estimates for nonlinear composites incorporating field fluctuations: I—theory, J. Mech. Phys. Solid. 50 (4) (2002) 737–757.
[7] B.L. Adams, M. Lyon, B. Henrie, Microstructures by design: linear problems in elastic–plastic design, Int. J. Plast. 20 (8–9) (2004) 1577–1602.
[8] S. Mercier, A. Molinari, Homogenization of elastic–viscoplastic heterogeneous materials: self-consistent and mori-tanaka schemes, Int. J. Plast. 25 (6) (2009) 1024–1048.
[9] C.N. Tomé, R.A. Lebensohn, Self consistent homogenization methods for texture and anisotropy, in: Continuum Scale Simulation of Engineering Materials: Fundamentals–microstructures–process Applications, 2004, pp. 473–499.
[10] V. Levin, F. Sabina, J. Bravo-Castillero, R. Guinovart-Diaz, R. Rodriguez-Ramos, O. Valdiviezo-Mijangos, Analysis of effective properties of electroelastic composites using the self-consistent and asymptotic homogenization methods, Int. J. Eng. Sci. 46 (8) (2008) 818–834.
[11] H. Garmestani, S. Lin, B. Adams, S. Ahzi, Statistical continuum theory for large plastic deformation of polycrystalline materials, J. Mech. Phys. Solid. 49 (3) (2001) 589–607.
[12] E. Kröner, Bounds for effective elastic moduli of disordered materials, J. Mech. Phys. Solid. 25 (2) (1977) 137–155.
[13] E. Kröner, Statistical modelling, in: Modelling Small Deformations of Polycrystals, Springer, 1986, pp. 229–291.
[14] H. Moulinec, P. Suquet, A numerical method for computing the overall response of nonlinear composites with complex microstructure, Comput. Methods Appl. Mech. Eng. 157 (1–2) (1998) 69–94.
[15] A. Prakash, R. Lebensohn, Simulation of micromechanical behavior of polycrystals: finite elements versus fast fourier transforms, Model. Simulat. Mater. Sci. Eng. 17 (6) (2009), 064010.
[16] R.A. Lebensohn, A.K. Kanjarla, P. Eisenlohr, An elasto-viscoplastic formulation based on fast fourier transforms for the prediction of micromechanical fields in polycrystalline materials, Int. J. Plast. 32 (2012) 59–69.
[17] Y.C. Yabansu, D.K. Patel, S.R. Kalidindi, Calibrated localization relationships for elastic response of polycrystalline aggregates, Acta Mater. 81 (2014) 151–160.
[18] Y.C. Yabansu, S.R. Kalidindi, Representation and calibration of elastic localization kernels for a broad class of cubic polycrystals, Acta Mater. 94 (2015) 26–35.
[19] R. Liu, Y.C. Yabansu, A. Agrawal, S.R. Kalidindi, A.N. Choudhary, Machine learning approaches for elastic localization linkages in high-contrast composite materials, Integrating Mater. Manuf. Innov. 4 (1) (2015) 13.
[20] R. Liu, Y.C. Yabansu, Z. Yang, A.N. Choudhary, S.R. Kalidindi, A. Agrawal, Context aware machine learning approaches for modeling elastic localization in three-dimensional composite microstructures, Integrating Mater. Manuf. Innov. (2017) 1–12.
[21] G. Landi, S.R. Kalidindi, Thermo-elastic localization relationships for multiphase composites, Comput. Mater. Continua (CMC) 16 (3) (2010) 273–293.
[22] G. Landi, S.R. Niezgoda, S.R. Kalidindi, Multi-scale modeling of elastic response of three-dimensional voxel-based microstructure datasets using novel dft-based knowledge systems, Acta Mater. 58 (7) (2010) 2716–2725.
[23] T. Fast, S.R. Kalidindi, Formulation and calibration of higher-order elastic localization relationships using the mks approach, Acta Mater. 59 (11) (2011) 4595–4605.
[24] M.W. Priddy, N.H. Paulson, S.R. Kalidindi, D.L. McDowell, Strategies for rapid parametric assessment of microstructure-sensitive fatigue for hcp polycrystals, Int. J. Fatig. 104 (2017) 231–242.
[25] C. Przybyla, R. Prasannavenkatesan, N. Salajegheh, D.L. McDowell, Microstructure-sensitive modeling of high cycle fatigue, Int. J. Fatig. 32 (3) (2010) 512–525.
[26] V. Shankar, M. Valsan, K.B.S. Rao, R. Kannan, S. Mannan, S. Pathak, Low cycle fatigue behavior and microstructural evolution of modified 9cr–1mo ferritic steel, Mater. Sci. Eng., A 437 (2) (2006) 413–422.
[27] Y. Djebara, A. El Moumen, T. Kanit, S. Madani, A. Imad, Modeling of the effect of particles size, particles distribution and particles number on mechanical properties of polymer-clay nano-composites: numerical homogenization versus experimental results, Compos. B Eng. 86 (2016) 135–142.
[28] V. Kouznetsova, M. Geers, W. Brekelmans, Multi-scale second-order computational homogenization of multi-phase materials: a nested finite element solution strategy, Comput. Methods Appl. Mech. Eng. 193 (48–51) (2004) 5525–5550.
[29] N.H. Paulson, M.W. Priddy, D.L. McDowell, S.R. Kalidindi, Reduced-order structure-property linkages for polycrystalline microstructures based on 2-point statistics, Acta Mater. 129 (2017) 428–438.
[30] D.J. Luscher, D.L. McDowell, C.A. Bronkhorst, A second gradient theoretical framework for hierarchical multiscale modeling of materials, Int. J. Plast. 26 (8) (2010) 1248–1275.
[31] J.E. Aarnes, S. Krogstad, K.-A. Lie, A hierarchical multiscale method for two-phase flow based upon mixed finite elements and nonuniform coarse grids, Multiscale Model. Simul. 5 (2) (2006) 337–363.
[32] R.A. Lebensohn, A.D. Rollett, P. Suquet, Fast fourier transform-based modeling for the determination of micromechanical fields in polycrystals, JOM (J. Occup. Med.) 63 (3) (2011) 13–18.
[33] T. Fast, S.R. Niezgoda, S.R. Kalidindi, A new framework for computationally efficient structure–structure evolution linkages to facilitate high-fidelity scale bridging in multi-scale materials models, Acta Mater. 59 (2) (2011) 699–707.
[34] M. Guillaumin, J. Verbeek, C. Schmid, Is that you? metric learning approaches for face identification, in: Computer Vision, 2009 IEEE 12th International Conference on, IEEE, 2009, pp. 498–505.
[35] C. Huang, S. Zhu, K. Yu, Large Scale Strongly Supervised Ensemble Metric Learning, with Applications to Face Verification and Retrieval, arXiv preprint arXiv:1212.6094.
[36] A. Mignon, F. Jurie, Pcca: a new approach for distance learning from sparse

pairwise constraints, in: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, IEEE, 2012, pp. 2666–2672.
[37] B. Ma, X. Ban, H. Huang, Y. Chen, W. Liu, Y. Zhi, Deep learning-based image segmentation for al-la alloy microscopic images, Symmetry 10 (4) (2018) 107.
[38] M.L. Silvoster, V. Govindan, Enhanced cnn based electron microscopy image segmentation, Cybern. Inf. Technol. 12 (2) (2012) 84–97.
[39] M. Ziatdinov, A. Maksov, S.V. Kalinin, Learning surface molecular structures via machine vision, npj Computational Materials 3 (1) (2017) 31.
[40] R. Liu, A. Agrawal, W.-k. Liao, A. Choudhary, M. De Graef, Materials discovery: understanding polycrystals from large-scale electron patterns, in: Big Data (Big Data), 2016 IEEE International Conference on, IEEE, 2016, pp. 2261–2269.
[41] R. Cang, H. Li, H. Yao, Y. Jiao, Y. Ren, Improving Direct Physical Properties Prediction of Heterogeneous Materials from Imaging Data via Convolutional Neural Network and a Morphology-aware Generative Model, arXiv preprint arXiv:1712.03811.
[42] R. Liu, L. Ward, C. Wolverton, A. Agrawal, W. Liao, A. Choudhary, Deep learning for chemical compound stability prediction, in: Proceedings of ACM SIGKDD Workshop on Large-scale Deep Learning for Data Mining, DL-KDD), 2016, pp. 1–7.
[43] Z. Yang, Y.C. Yabansu, R. Al-Bahrani, W.-k. Liao, A.N. Choudhary, S.R. Kalidindi, A. Agrawal, Deep learning approaches for mining structure-property linkages in high contrast composites from simulation datasets, Comput. Mater. Sci. 151 (2018) 278–287.
[44] A. Cecen, H. Dai, Y.C. Yabansu, L. Song, S.R. Kalidindi, Material structure-property linkages using three-dimensional convolutional neural networks, Acta Mater. 146 (2018) 76–84.
[45] R. Kondo, S. Yamakawa, Y. Masuoka, S. Tajima, R. Asahi, Microstructure recognition using convolutional neural networks for prediction of ionic conductivity in ceramics, Acta Mater. 141 (2017) 29–38.
[46] X. Li, Y. Zhang, H. Zhao, C. Burkhart, L. C. Brinson, W. Chen, A Transfer Learning Approach for Microstructure Reconstruction and Structure-property Predictions, arXiv preprint arXiv:1805.02784.
[47] R. Cang, Y. Xu, S. Chen, Y. Liu, Y. Jiao, M.Y. Ren, Microstructure representation and reconstruction of heterogeneous materials via deep belief network for computational material design, J. Mech. Des. 139 (7) (2017), 071404.
[48] Z. Yang, X. Li, L.C. Brinson, A. Choudhary, W. Chen, A. Agrawal, Microstructural materials design via deep adversarial learning methodology, J. Mech. Des. 140 (11) (2018) 10.
[49] X. Li, Z. Yang, L.C. Brinson, A. Choudhary, A. Agrawal, W. Chen, A deep adversarial learning methodology for designing microstructural material systems, in: ASME 2018 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, American Society of Mechanical Engineers, 2018. V02BT03A008–V02BT03A008.
[50] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial nets, in: Advances in Neural Information Processing Systems, 2014, pp. 2672–2680.
[51] B.L. Adams, X.C. Gao, S.R. Kalidindi, Finite approximations to the second-order properties closure in single phase polycrystals, Acta Mater. 53 (13) (2005) 3563–3577.
[52] S.R. Kalidindi, S.R. Niezgoda, G. Landi, S. Vachhani, T. Fast, A novel framework for building materials knowledge systems, Comput. Mater. Continua (CMC) 17 (2) (2010) 103–125.
[53] S.R. Niezgoda, A.K. Kanjarla, S.R. Kalidindi, Novel microstructure quantification framework for databasing, visualization, and analysis of microstructure data, Integrating Mater. Manuf. Innov. 2 (1) (2013) 3.
[54] P. Altschuh, Y.C. Yabansu, J. Hötzer, M. Selzer, B. Nestler, S.R. Kalidindi, Data science approaches for microstructure quantification and feature identification in porous membranes, J. Membr. Sci. 540 (2017) 88–97.
[55] A. Iskakov, Y.C. Yabansu, S. Rajagopalan, A. Kapustina, S.R. Kalidindi, Application of spherical indentation and the materials knowledge system framework to establishing microstructure-yield strength linkages from carbon steel scoops excised from high-temperature exposed components, Acta Mater. 144 (2018) 758–767.
[56] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proc. IEEE 86 (11) (1998) 2278–2324.
[57] H. Schulz, S. Behnke, Learning object-class segmentation with convolutional neural networks, in: ESANN, 2012.
[58] F. Ning, D. Delhomme, Y. LeCun, F. Piano, L. Bottou, P.E. Barbano, Toward automatic phenotyping of developing embryos from videos, IEEE Trans. Image Process. 14 (9) (2005) 1360–1371.
[59] P. Sermanet, Y. LeCun, Traffic sign recognition with multi-scale convolutional networks, in: Neural Networks (IJCNN), the 2011 International Joint Conference on, IEEE, 2011, pp. 2809–2813.
[60] D.C. Ciresan, U. Meier, J. Masci, L. Maria Gambardella, J. Schmidhuber, Flexible, high performance convolutional neural networks for image classification, in: IJCAI Proceedings-international Joint Conference on Artificial Intelligence, vol. 22, Barcelona, Spain, 2011, p. 1237.
[61] T. Wang, D.J. Wu, A. Coates, A.Y. Ng, End-to-end text recognition with convolutional neural networks, in: Pattern Recognition (ICPR), 2012 21st International Conference on, IEEE, 2012, pp. 3304–3308.
[62] S. Ji, W. Xu, M. Yang, K. Yu, 3d convolutional neural networks for human action recognition, IEEE Trans. Pattern Anal. Mach. Intell. 35 (1) (2013) 221–231.
[63] V. Nair, G.E. Hinton, Rectified linear units improve restricted Boltzmann machines, in: Proceedings of the 27th International Conference on Machine Learning, ICML-10), 2010, pp. 807–814.
[64] B.L. Adams, S. Kalidindi, D.T. Fullwood, Microstructure-sensitive Design for Performance Optimization, Butterworth-Heinemann, 2013.
[65] X. Glorot, Y. Bengio, Understanding the difficulty of training deep feedforward neural networks, Aistats 9 (2010) 249–256.
[66] D. Kingma, J. Ba, Adam: A Method for Stochastic Optimization, arXiv preprint arXiv:1412.6980.
[67] Karlsson Hibbett, Sorensen, ABAQUS/standard: User's Manual, vol. 1, Hibbitt, Karlsson & Sorensen, 1998.
[68] H.F. Al-Harbi, G. Landi, S.R. Kalidindi, Multi-scale modeling of the elastic response of a structural component made from a composite material using the materials knowledge system, Model. Simulat. Mater. Sci. Eng. 20 (5) (2012), 055001.
[69] S.R. Kalidindi, G. Landi, D.T. Fullwood, Spectral representation of higher-order localization relationships for elastic behavior of polycrystalline cubic materials, Acta Mater. 56 (15) (2008) 3843–3853.
[70] F. Chollet, Keras, 2015. https://github.com/fchollet/keras.
[71] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G.S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, X. Zheng, TensorFlow: Large-scale Machine Learning on Heterogeneous Systems, 2015 software available from: tensorflow.org, http://tensorflow.org/.
[72] 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.
[73] D. Wheeler, D. Brough, T. Fast, S. Kalidindi, A. Reid, Pymks, Materials Knowledge System in python, figshare, 2014, 2016.
[74] S.J. Pan, Q. Yang, A survey on transfer learning, IEEE Trans. Knowl. Data Eng. 22 (10) (2010) 1345–1359.
[75] R. Liu, A. Kumar, Z. Chen, A. Agrawal, V. Sundararaghavan, A. Choudhary, A predictive machine learning approach for microstructure optimization and materials design, Sci. Rep. 5.
[76] B. Meredig, A. Agrawal, S. Kirklin, J.E. Saal, J. Doak, A. Thompson, K. Zhang, A. Choudhary, C. Wolverton, Combinatorial screening for new materials in unconstrained composition space with machine learning, Phys. Rev. B 89 (9) (2014), 094104.
[77] A. Agrawal, A. Choudhary, Perspective: materials informatics and big data: realization of the “fourth paradigm” of science in materials science, Apl. Mater. 4 (5) (2016), 053208.
[78] P. Fernandez-Zelaia, V.R. Joseph, S.R. Kalidindi, S.N. Melkote, Estimating mechanical properties from spherical indentation using bayesian approaches, Mater. Des. 147 (2018) 92–105.
[79] D.K. Patel, S.R. Kalidindi, Estimating the slip resistance from spherical nanoindentation and orientation measurements in polycrystalline samples of cubic metals, Int. J. Plast. 92 (2017) 19–30.
[80] D.K. Patel, H.F. Al-Harbi, S.R. Kalidindi, Extracting single-crystal elastic constants from polycrystalline samples using spherical nanoindentation and orientation measurements, Acta Mater. 79 (2014) 108–116.