IJCNN 2019. International Joint Conference on Neural Networks. Budapest, Hungary. 14-19 July 2019

Deep learning based domain knowledge integration for small datasets: Illustrative applications in materials informatics

Zijiang Yang*, Reda Al-Bahrani*, Andrew C. E. Reid†,
Stefanos Papanikolaou‡, Surya R. Kalidindi§, Wei-keng Liao*, Alok Choudhary*, and Ankit Agrawal*
*Department of Electrical Engineering and Computer Science, Northwestern University
†Center for Theoretical and Computational Materials Science, National Institute of Standards and Technology
‡Department of Mechanical & Aerospace Engineering, West Virginia University
§George W. Woodruff School of Mechanical Engineering, Georgia Institute of Technology
*{zyz293, rav650, wkliao, choudhar, ankitag}@eecs.northwestern.edu,
†andrew.reid@nist.gov, ‡stefanos.papanikolaou@mail.wvu.edu, §surya.kalidindi@me.gatech.edu

***Abstract*—Deep learning has shown its superiority to traditional machine learning methods in various fields, and in general, its success depends on the availability of large amounts of reliable data. However, in some scientific fields such as materials science, such big data is often expensive or even impossible to collect. Thus given relatively small datasets, most of data-driven methods are based on traditional machine learning methods, and it is challenging to apply deep learning for many tasks in these fields. In order to take the advantage of deep learning even for small datasets, a domain knowledge integration approach is proposed in this work. The efficacy of the proposed approach is tested on two materials science datasets with different types of inputs and outputs, for which domain knowledge-aware convolutional neural networks (CNNs) are developed and evaluated against traditional machine learning methods and standard CNN-based approaches. Experiment results demonstrate that integrating domain knowledge into deep learning can not only improve the model's performance for small datasets, but also make the prediction results more explainable based on domain knowledge.**

I. Introduction

Deep learning has led to a series of breakthroughs in various fields, such as computer vision [1], [2] and natural language processing [3], [4]. One of the most important factors that is believed to be responsible for the success of deep learning is the availability of large amounts of reliable data. With flexible architecture and striking learning capability, deep learning models have the ability to extract crucial information from such huge amounts of data effectively and automatically. Thus, deep learning has gained significant attention in other scientific research fields, such as biology [5], [6], chemistry [7] and materials science [8]–[10]. However, for some research problems in these fields, it is difficult or even impossible to collect a large amounts of data for the application of deep learning, because data comes from some expensive and time-consuming experiments or simulations. Meanwhile, deep learning has obvious drawbacks working on small datasets, such as overfitting. In addition, it is also difficult to further improve the performance of deep learning models due to limited data. In order to still take advantage of deep learning and achieve good performance training on small datasets, two commonly used solutions are applied in this situation.

The first solution is transfer learning, which is widely used in the research of convolutional neural network (CNN) [11]. Transfer learning is a technique that uses the knowledge learned from a huge dataset to help the training process of deep learning models on the other small datasets. The earlier layers of a CNN can learn low-level simple features such as lines or curves, while subsequent layers combine them into high-level problem-specific features to solve certain problems. The simple features learned in earlier layers are usually shared among different datasets. Thus, transfer learning can achieve relatively good performance even if the two datasets are dissimilar. Transfer learning has been successfully employed in abdominal ultrasound images classification [12], pavement crack damage detection [9], and large-scale protein subcellular localization [13]. However, in some fields, such as materials science, transfer learning might not always be the best solution. This is because the microstructures of materials systems are significantly different from natural objects that pre-trained models of transfer learning are usually trained on. It can therefore be tricky to transfer the knowledge learned by pre-trained models. Moreover, the lack of data hinders the development of pre-trained models specifically for such fields.

The other solution is to integrate domain knowledge into deep learning to guide the training process so that it can achieve good performance even with relatively small datasets. In [14], Tang et al. design a deep convolutional network combining domain knowledge and the representation ability of deep learning for fingerprint minutiae extraction. In [6], deep learning is combined with domain knowledge to detect gland in H&E histology tissue images and it improves the state-of-the-art results. In [15], the use of biologically plausible rules in neural network architectures is explored to learn view-invariant features of human faces. Thus, there is emerging evidence that by appropriately integrating domain knowledge, deep learning

978-1-7281-2009-6/$31.00 paper N-19941.pdf

can be adjusted to solve specific problems and improve the performance for certain problems.

In this work, we propose a deep learning based domain knowledge integration approach on small datasets and apply it to solve two prediction problems in materials science. More specifically, two-point correlation function, which is used to capture the spatial information from data, is used as the domain knowledge to be integrated in the training of the deep learning model. In order to evaluate the performance of the proposed method, two different datasets are used, one is homogenization dataset and the other is crystal plasticity dataset. The proposed hybrid CNN with domain knowledge integration is compared against traditional machine learning methods, CNNs without domain knowledge integration as well as against CNN with naive domain knowledge integration. The performance improvement in the experimental results demonstrates that domain knowledge can be properly integrated into deep learning models. Moreover, the prediction results are more physically explainable by integrating domain knowledge. Though the experimental results are for two materials science datasets, this approach can be easily extended to other scientific fields.

II. Background and Related Work

In section II-A, two-point correlation function (i.e., domain knowledge) is introduced. Then, we define the problems for the two applications used in this work in sections II-B and II-C, respectively.

A. Two-point Correlation Function

N-point correlation function can fully capture the spatial information from the data [16]. However, considering the trade off between the computational cost and accuracy, two-point correlation function is usually used in practice. Two-point correlation function (also known as two-point statistics) presents the probability of a vector with a given length and orientation falling in two specific local states presented in the image. In other words, two-point correlation function gives the information about how local states are spatially distributed in the image. Figure 1 shows a discretized microstructure (i.e. a binary image) illustrating the concepts of pixel location u, vector r and local state s in equations 1 and 2, which are equations for computing two-point correlation function,

$$f(r|s,s') = \frac{1}{U}\sum_{u} I(u,s)I(u+r,s') \tag{1}$$

$$I(u,s) = \begin{cases} 1 & \text{if local state at } u \text{ is } s \\ 0 & \text{else} \end{cases} \tag{2}$$

In these equations, s and s' denote two specific local states in the image. u is the pixel location in the image, and U is the total number of pixel locations in the image. $I(u,s)$ is the indicator function. Thus, $f(r|s,s')$ calculates the conditional probability of finding the local state s and s' at a distance and orientation away from each other defined by the vector r. When the two local states are the same (i.e. $s = s'$), it is referred to as a two-point auto-correlation. Otherwise, it is referred to as a two-point cross-correlation.

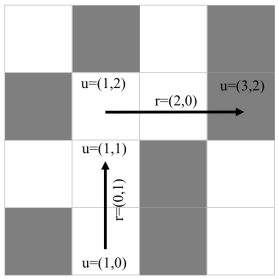

Fig. 1. Illustration of the discretized microstructure. It can be considered as a binary image representing two local states of the microstructure. The value of pixel is either 0 or 1 representing the black or white local state. The origin pixel is the left-bottom pixel indexed from 0. For example, the local state of pixel location $(1,0)$ (i.e. $u = (1,0)$) is 1 (i.e. $s = 1$).

Two-point correlation function are widely used as features to train traditional machine learning models in materials science [17], [18], astronomy and cosmology [19], [20]. In this work, two-point auto-correlation is integrated in the deep learning model so that domain knowledge can help the training process of the deep learning given a small dataset resulting in a performance improvement.

B. Macroscale (effective) Stiffness Prediction for Homogenization Linkages

Processing-structure-property (PSP) linkages of materials systems are crucial in order to understand the role of the materials internal structure in controlling its properties, and discover new innovative materials. In other words, there is a complex inherent relationship between material structure and property of materials. Only when we fully understand such relationship can we accurately calculate materials' properties according to their structures and effectively design new materials given desired properties. Homogenization is to understand materials' inherent relationship from lower length scale to higher length scale (e.g., given microscale structural information, what is the overall macroscale property of the material). Some related work has already been done to model homogenization linkages. In [17], [18], two-point auto-correlation is computed for the microstructures, and principal component analysis (PCA) is applied to obtain the reduced-order representations, and finally standard regression method is implemented to predict the property of materials. In [21], Yang et al. develop a stand-alone convolutional neural network to model the homogenization of high-contrast composite materials.

C. Crystal Plasticity Prediction Using Digital Image Correlation

Detection of prior deformation history (i.e. plasticity history) of a material is important in industrial applications. This is because the mechanical properties of materials could be dramatically deteriorated if they have been heavily deformed in the past, even though the chemical composition of the

materials remains the same. Digital Image Correlation (DIC) [22] is an inexpensive and flexible method for this problem. More specifically, the calculation of the DIC includes several steps. First, a series of gray scale images representing different deformation stages of the materials are compared. Then the movement of pixels in a representative volume is tracked. Finally, a correlation algorithm is used to calculate the corresponding displacement and strain. In [23], Papanikolaou et al. use the reduced-order representations of two-point autocorrelation of dislocation microstructures as features to train a machine learning model, and find a clear cluster formation of samples based on their responses to the DIC reloading strain.

III. Datasets

A. Homogenization Dataset

The homogenization dataset[1] is generated by micromechanical finite element models. The dataset includes three-dimensional microstructures of high-contrast two-phase composite materials, and each microstructure is referred to as a microscale volume element (MVE) (as shown in Figure 2 (a)). More specifically, each MVE can be considered as a $51 \times 51 \times 51$ binary image and there are 8550 MVEs totally to ensure a rich morphological diversity of microstructures. Meanwhile, there are 50 categories of volume fractions (i.e., the fraction of one of the two phases in a MVE) ranging from 25% to 75%, and each volume fraction category contains 171 MVEs. For each volume fraction category, 114 MVEs are selected for training and the rest for testing. A validation set is created by randomly selecting 33% of the MVEs for the training process to tune the hyperparameters for the deep learning model. In other words, 8550 MVEs are split into 3 sets where training set includes 3819 MVEs, validation set has 1881 MVEs and testing set contains 2850 MVEs. The response of each MVE is its macroscale (effective) stiffness, which is a continuous real value representing the material property of interest. Thus, a regression model is developed for this dataset.

B. Crystal Plasticity Dataset

Crystal plasticity dataset[1]is generated by 2D discrete dislocation dynamics (DDD) simulation, which is a simulation used to emulate DIC. Each data point in this dataset is a strain profile in 2D-DDD simulations (as shown in Figure 2 (b)) and there are three variables in the dataset, which are sample width, reload-strain and slip type of materials systems. There are six different sample widths, two reload-strains and two slip types resulting in 24 subsets of data, which are shown in Table I. More specifically, each data point can be considered as a two-dimensional one-channel image and the values of pixels are real values representing the local strain. However, the dimensions of images are varied for different subsets, which increases as a factor of the sample width, while the dimensions of images in the same subset can also be slightly different. As an example, for a data point with a width of 0.0625 μm the dimension of the image is around 30×125, while for another data point with a width of 0.125 μm the image is around 60×250 pixels. In order to ensure that the testing set contains representations from all the subsets, approximately 25% of the data are selected as testing set for each response category in each subset (stratified sampling). For the rest of the data, approximately 20% are randomly selected as the validation set. In other words, the training set contains 1262 data points, the validation set has 316 data points, and the testing set includes 504 data points. The response of each data point is its initial loading strain, which are categorical (i.e. 0.1%, 1% or 10%). Thus, a classification model is established to solve this problem.

[1]dataset is available on request

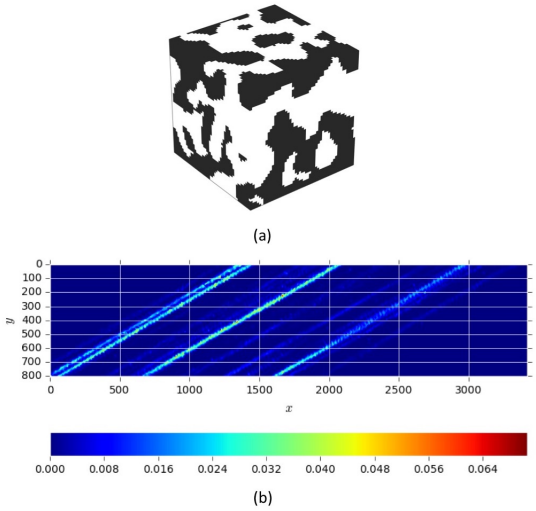

Fig. 2. Data example for homogenization dataset and crystal plasticity dataset. (a) a MVE from the homogenization dataset, and it can be considered as a 3D binary image (i.e. 0 and 1 represent hard phase and soft phase, respectively.) (b) a strain profile from the crystal plasticity dataset. It can be considered as a 2D one-channel image and the value of each pixel is a continuous real value representing the local strain.

TABLE I
Number of Data Points for Each Subset in Crystal Plasticity Dataset

Width (μm)	**2**	**1**	**0.5**	**0.25**	**0.125**	**0.0625**
Small-reload & two slips	146	147	145	147	147	147
Small-reload & one slip	27	27	27	27	27	27
Large-reload & two slips	146	147	145	147	147	147
Large-reload & one slip	27	27	27	27	27	27

IV. Methods

Data preprocessing is introduced in section IV-A, and the standard CNN without domain knowledge integration is presented in section IV-B. Then the hybrid CNN with domain knowledge is proposed in section IV-C.

A. Preprocessing

1) Homogenization dataset: Each MVE can be considered as a $51 \times 51 \times 51$ binary image, which means the values of elements in MVE are either zeros (i.e. hard phase) or ones (i.e. soft phase). Since, large amounts of zeros in the input might significantly deteriorate the functionality of convolution operations of the CNN, the values of the input are rescaled from [0, 1] to [-0.5, 0.5].

2) Crystal plasticity dataset: This dataset is relatively small for training the deep learning model, so an aggressive cropping approach is applied to augment the dataset.

- We resize the image to 4 scales where the dimensions are [256, 768], [288, 864], [320, 960] and [352, 1056], respectively.
- For each resized image, we take the left, center and right squares.
- For each square, we take the center 224×224 crop as well as the square resized to 224×224.
- Because the input data in different subsets have different scales, we normalize the input data, which means the input data is first zero-centered, then divided by its standard deviation.

Figure 3 illustrates the first three preprocessing steps. This cropping approach leads to 24 (i.e. $4 \times 3 \times 2$) crops per image, which enlarges the dataset 24 times. Note that in the testing time, the softmax probability of the final prediction for each image is averaged over the probabilities of all of its crops.

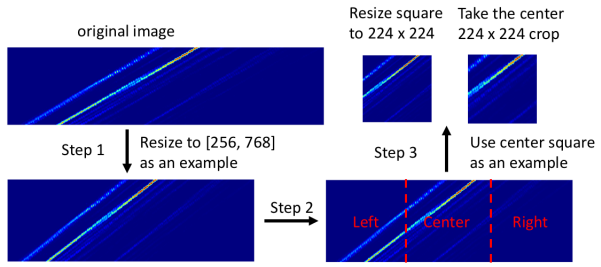

Fig. 3. Illustration of the first three preprocessing steps of crystal plasticity dataset. As an example, original image is resized to [256, 768] in step 1 and center square is used in step 3 in this figure.

B. Standard CNN Architecture

Different architectures of varying depths and different number of filters in each convolution layer are explored to examine which architecture produces the best standard CNN for each application. Then, a greedy approach is used to search for the best combination of hyperparameters of selected architecture.

1) Homogenization dataset: In [21], a classic CNN is developed to model homogenization linkage, and its configuration is shown in Figure 4 (a). In this work, we use the same CNN as "standard CNN", for comparing with the proposed hybrid approach. More specifically, the above-mentioned deep learning model is a 3D CNN, which takes a $51 \times 51 \times 51$ binary image as input and produces a continuous real value to predict the effective stiffness of the input material microstructure. There are five convolutional layers with $3 \times 3 \times 3$ filters, and the number of filters in each convolutional layer is 16, 32, 64, 128 and 256, respectively. $2 \times 2 \times 2$ average pooling layers are used to reduce dimensionality. Finally, the CNN has two fully connected layers with 2048 and 1024 neurons, respectively. Rectified Linear Unit (ReLU) [24] is used as activation function for each convolutional layer and fully connected layer, except for the output layer a linear activation function is used to produce continuous real values. The normalized initialization method [25] is used to initialize the weights of the CNN. In order to avoid overfitting, $L2$ regularization with penalty factor as 0.001 is applied in each convolutional layer and fully connected layers. Moreover, early stopping is used to further avoid overfitting. The training process is terminated if the value of the loss function on the validation set does not improve for 10 epochs. As for the optimizer, Adam [26] with learning rate as 0.001, β_1 as 0.9 and β_2 as 0.999 is used. The batch size is 32 MVEs for training.

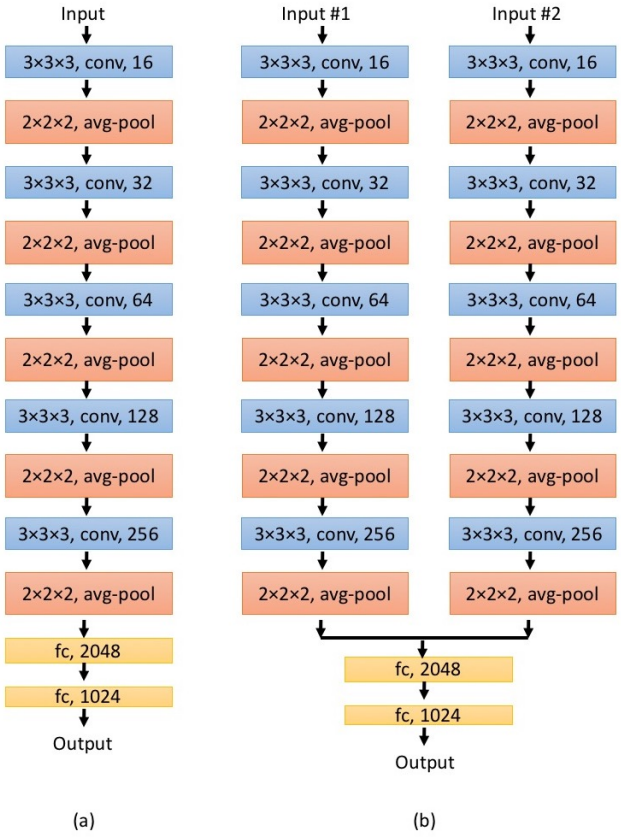

Fig. 4. Configuration of CNNs for homogenization dataset (a) standard CNN from [21]; (b) proposed hybrid CNN.

2) Crystal plasticity dataset: A CNN with state-of-the-art techniques such as residual module [1] and batch normalization [27] are implemented to predict initial loading strain of the material, and its configuration is shown in Figure 5 (a). This deep learning model is a 2D CNN, which takes 224×224 one-channel image as input and classifies it into one of three initial loading strains. The first convolutional

layer has 16 3×3 filters, followed by a 3×3 max pooling layer with strides 2. Then there are two residual modules, and each has four convolutional layers. Each convolutional layer in the two residual modules has 32 and 64 3×3 filters, respectively. The convolutional layer for identity mapping in the two residual module has 32 and 64 1×1 filters, respectively. A 3×3 max pooling layer with strides 2 is followed by each residual module to reduce dimensionality. Finally, global average pooling layer [28] is used. Note that batch normalization is used for each convolutional layer, and it is applied before the activation functions. ReLU is used as the activation function for each convolutional layer, while softmax activation function is used in the output layer to make the classification. Early stopping is applied to avoid overfitting where the training process is stopped if loss on validation set does not improve for 20 epochs. Weights initialization method, optimizer and batch size are the same as the homogenization dataset.

C. Hybrid CNN Architecture to Integrate Domain Knowledge

In this section, we describe the development of the proposed hybrid CNN, building on the standard CNN, to integrate the domain knowledge for each dataset. In general, the hybrid CNN has two identical sub-branches containing the convolutional and pooling part of its standard CNN, and the outputs of two sub-branches are concatenated together to be fed into the fully connected layers to produce final predictions. Particularly, the input for each sub-branch can be either original data or its two-point auto-correlation. The configuration of each hybrid CNN is introduced in the following sections.

1) Homogenization dataset: The hybrid CNN is presented in Figure 4 (b). This CNN includes two identical sub-branches that contain the convolutional and pooling layers of the corresponding standard CNN. After the fifth average pooling layer, the feature maps of two sub-branches are flattened and concatenated together to form a one-dimensional vector. Then, this vector is fed into two fully connected layers with 2048 and 1024 neurons, one after the other. Note that the other hyperparameter settings of this hybrid CNN are the same as its standard CNN.

2) Crystal plasticity dataset: Figure 5 (b) presents the hybrid CNN for crystal plasticity dataset. This hybrid CNN also has two sub-branches that contain all the layers before the output layer. The outputs of two sub-branches after global average pooling layer are concatenated together, followed by two fully connected layers with 256 and 128 neurons, one after the other. In addition, batch normalization is applied before the ReLU activation function for both fully connected layers. Dropout layer is used after each fully connected layer with dropout rate as 0.3 and 0.2, respectively. Moreover, other hyperparameter settings of this hybrid CNN are the same as its standard CNN.

V. Results and Discussion

In order to integrate domain knowledge, two-point auto-correlation is first computed for each image data in the two

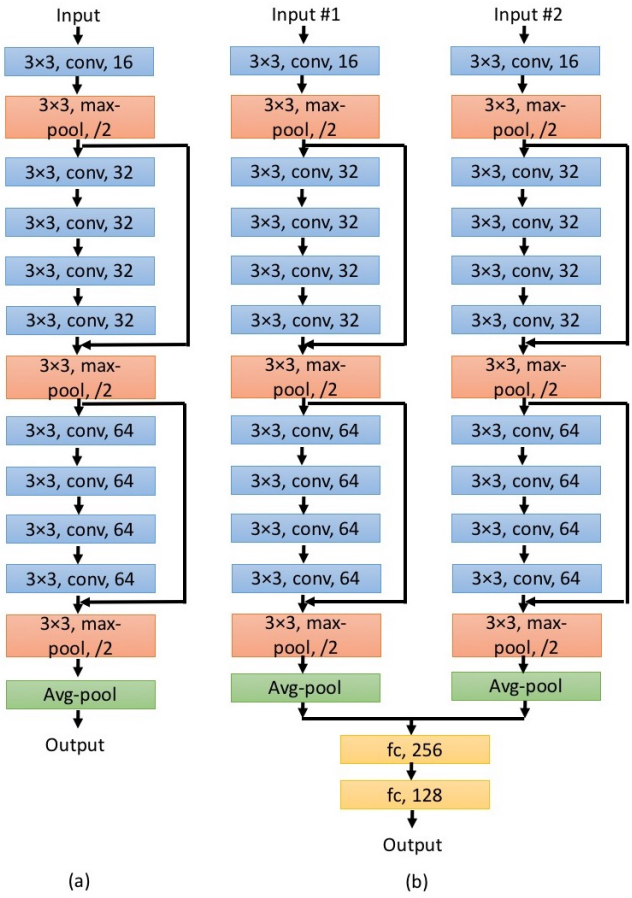

Fig. 5. Configuration of CNNs for crystal plasticity dataset. (a) standard CNN; (b) proposed hybrid CNN.

datasets. Note that the image data has the same dimensionality as its two-point auto-correlation. Thus, the proposed method is implemented by a hybrid CNN, which takes the original image data as well as its corresponding two-point auto-correlation as input for the two sub-branches, respectively (referred to as hybrid CNN No.3). For each of the two applications in this work, the proposed method is compared with seven benchmarks. The seven benchmarks are categorized into three groups.

1) Traditional machine learning methods: The first two benchmarks are two traditional machine learning models. In traditional machine learning methods, PCA is applied on the two-point auto-correlation of original image data to obtain reduced-order representations, which are subsequently fit to property values using Random Forest and Gradient Boosting methods, respectively. More specifically, the Random Forest is defined as a classifier consisting of 100 decision tree classifiers, and Gradient boosting uses regression trees as base learners with 100 training iterations.
2) Standard CNN: Another three benchmarks are implemented by standard CNN. Standard CNN No.1 takes the original image data as input, and standard CNN

No.2 takes the two-point auto-correlation of the original image data as the input. In addition, the two-point auto-correlation is used as an additional channel of the original image data, and standard CNN No.3 takes it as the input.

3) Hybrid CNN: Another two experiments are carried out using the hybrid CNN as two benchmarks. Hybrid CNN No.1 takes the original image data as input for both sub-branches, while Hybrid CNN No.2 takes the two-point auto-correlation of the original image data as input for both sub-branches.

A. Results: Homogenization Dataset

In [21], CNN is trained to model the homogenization linkages and the performance is evaluated in terms of mean absolute stiffness error (MASE). The calculation of the MASE consists of two steps. Firstly, the mean absolute error between predicted values and ground truth values are computed. Then this mean absolute error is divided by the mean of effective stiffness of the entire testing set to get the MASE. In order to make a fair comparison, we use the same dataset, the same training/testing split, and the same error metric to compare the model's performance.

Table II lists the performance for each method. We can observe that traditional machine learning models achieve 17.26% and 16.88% MASE, while the standard CNN No.1, No.2 and No.3 obtain 3.10%, 7.67% and 4.28% MASE, respectively. On the other hand, hybrid CNN No.3 improves the MASE to 2.76%, while the MASE of hybrid CNN No.1 and No.2 decreases to 3.47% and 5.44%, respectively. In other words, by integrating domain knowledge, model's performance is improved by 11.0% (i.e. 1 - 2.76/3.10) compared to the best benchmark method. The results of standard CNN No.2 and hybrid CNN No.2 show that purely using two-point auto-correlation as input does not improve the performance compared with CNNs taking original image as input, because two-point auto-correlation can not fully capture the spatial information from the data [16]. On the other hand, two-point auto-correlation can indeed highlight some significant spatial information, and when it is integrated into the deep learning model in a proper way it can guide the training of the deep learning model for performance improvement. Thus, the result of the proposed hybrid CNN No.3 shows the best model's performance, while naively integrating two-point auto-correlation into the deep learning model in standard CNN No.3 does not improve the performance. In addition, since hybrid CNN No.1 and No.2 use the same configuration as hybrid CNN No.3 but deteriorate the model's performance, it shows that the superior performance of hybrid CNN No.3 is due to proper domain knowledge integration that can guide the training process of the deep learning model instead of simply increasing model's complexity.

TABLE II
PERFORMANCE COMPARISON OF DIFFERENT METHODS FOR HOMOGENIZATION DATASET

Methods	MASE
Random Forest	17.26%
Gradient Boosting	16.88%
Standard CNN No.1 [21]	3.10%
Standard CNN No.2	7.67%
Standard CNN No.3	4.28%
Hybrid CNN No.1	3.47%
Hybrid CNN No.2	5.44%
Hybrid CNN No.3	**2.76%**

B. Results: Crystal Plasticity Dataset

In [23], two-point auto-correlation is used as features to implement clustering for each subset of crystal plasticity dataset. However, the machine learning model is individually developed on each subset of data instead of the entire dataset, which can deteriorate model's generalization. In this work, we move forward for this problem by developing classification models directly on the entire dataset.

Table III presents the performance of the proposed method and benchmarks. Specifically, traditional machine learning models achieve 64.09% and 68.45% classification accuracy, while standard CNN No.1, No.2 and No.3 obtain 93.25%, 33.33% and 84.13% classification accuracy, respectively. On the other hand, classification accuracy is improved to 95.04% by hybrid CNN No.3, while performance of hybrid CNN No.1 and No.2 decrease to 93.06% and 33.33%, respectively. In other words, the model's performance is improved by 28.5% (i.e. 1 - 4.96/6.94) by integrating domain knowledge compared to the best benchmark method. The results support the conclusion that the performance improvement of the deep learning model is due to the proper integration of domain knowledge instead of simply increasing the model's complexity, and purely using two-point auto-correlation or naively integrating two-point auto-correlation into the deep learning model does not improve the model's performance.

TABLE III
PERFORMANCE COMPARISON OF DIFFERENT METHODS FOR CRYSTAL PLASTICITY DATASET

Methods	Accuracy
Random Forest	68.45%
Gradient Boosting	64.09%
Standard CNN No.1	93.25%
Standard CNN No.2	33.33%
Standard CNN No.3	84.13%
Hybrid CNN No.1	93.06%
Hybrid CNN No.2	33.33%
Hybrid CNN No.3	**95.04%**

Moreover, the prediction results are more physically explainable by integrating domain knowledge. Table IV and Table V show the classification accuracy of each subset data for crystal plasticity dataset based on standard CNN No.1 and hybrid CNN No.3, respectively. Empirically, data with large sample width and small-reload should be easier to predict [23]. Thus, the classification accuracy in two tables should have a general trend that accuracy goes down from top to bottom

and from left to right, which means bottom right corners in the two tables should have the worst classification accuracy. We can observe that Table V generally follows this trend and all the worst classification accuracies appear in the bottom right corner. However, Table IV does not have a clear trend, and some results do not match our experience. For instance, data with a width of 2 μm, large-reload & one slip achieves a relatively bad classification accuracy, while data with a width of 0.0625 μm, large-reload & one slip achieves a relatively good classification accuracy.

To visually demonstrate the benefit of integrating domain knowledge into the deep learning model, we extract the outputs of the second fully connected layer of hybrid CNN No.3 and plot the projection of data on first two principal components of these outputs in Figure 6. For large width samples (e.g. 2 μm and 1 μm, which are first two rows in Figure 6), samples are clearly separated except that there are small overlaps in the large-reload & two slips plots (i.e. second column of first two rows in Figure 6). When the sample width becomes smaller, the distance between clusters becomes smaller. For the samples of large-reload & one slip and large-reload & two slips with a width of 0.0625 μm (i.e. first two columns of the last row in Figure 6), red and yellow samples (i.e. samples with 0.1 % and 1 % strain prior loaded state) are significantly overlapped, which matches our intuition that samples with small width and large-reload are harder to predict. Thus, the results show that integrating domain knowledge not only improves the model's performance, but also makes the prediction results of the deep learning model amenable to better physical explanation.

VI. Conclusions

In this work, we developed a deep learning based domain knowledge integration approach on small datasets and apply it to solve two research problems in the materials science field. More specifically, two different small datasets are used in this work. Homogenization dataset contains three-dimensional binary MVEs as inputs and continuous real values as outputs, while crystal plasticity dataset includes two-dimensional one-channel images with different sizes as inputs and categorical values as outputs. To integrate domain knowledge, the proposed hybrid CNN is trained by taking the original image data as well as its two-point auto-correlation as inputs for its two sub-branches in both applications. By comparing against the traditional machine learning methods solely based on domain knowledge, standard CNN and hybrid CNNs that are solely based on either the original data or two-point auto-correlation as well as standard CNN that are based on naive domain knowledge integration, we conclude that given a small dataset, the proposed domain knowledge integration method can guide the training process of the deep learning model, which can improve the model's performance as well as make the prediction results more explainable with domain knowledge. Though we only evaluate the proposed method on two materials science related datasets, the proposed method can be easily extended to other scientific fields. Thus the proposed method could help the applications of deep learning in scientific research fields where large amounts of reliable data is not available.

Acknowledgment

This work is supported in part by the following grants: NIST award 70NANB14H012, NSF award CCF-1409601; DOE awards DE-SC0014330, DE-SC0019358.

References

[1] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[3] R. Collobert and J. Weston, "A unified architecture for natural language processing: Deep neural networks with multitask learning," in *Proceedings of the 25th international conference on Machine learning*. ACM, 2008, pp. 160–167.

[4] X. Glorot, A. Bordes, and Y. Bengio, "Domain adaptation for large-scale sentiment classification: A deep learning approach," in *Proceedings of the 28th international conference on machine learning (ICML-11)*, 2011, pp. 513–520.

[5] C. Angermueller, T. Pärnamaa, L. Parts, and O. Stegle, "Deep learning for computational biology," *Molecular systems biology*, vol. 12, no. 7, p. 878, 2016.

[6] J. Wang, J. D. MacKenzie, R. Ramachandran, and D. Z. Chen, "Detection of glands and villi by collaboration of domain knowledge and deep learning," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2015, pp. 20–27.

[7] A. Lusci, G. Pollastri, and P. Baldi, "Deep architectures and deep learning in chemoinformatics: the prediction of aqueous solubility for drug-like molecules," *Journal of chemical information and modeling*, vol. 53, no. 7, pp. 1563–1575, 2013.

[8] A. Agrawal and A. Choudhary, "Perspective: Materials informatics and big data: Realization of the fourth paradigm of science in materials science," *APL Materials*, vol. 4, no. 053208, pp. 1–10, 2016.

[9] K. Gopalakrishnan, H. Gholami, A. Vidyadharan, A. Choudhary, and A. Agrawal, "Crack damage detection in unmanned aerial vehicle images of civil infrastructure using pre-trained deep learning model," *International Journal for Traffic and Transport Engineering*, 2018, to appear.

[10] Z. Yang, X. Li, L. C. Brinson, A. N. Choudhary, W. Chen, and A. Agrawal, "Microstructural materials design via deep adversarial learning methodology," *arXiv preprint arXiv:1805.02791*, 2018.

[11] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[12] P. M. Cheng and H. S. Malhi, "Transfer learning with convolutional neural networks for classification of abdominal ultrasound images," *Journal of digital imaging*, vol. 30, no. 2, pp. 234–243, 2017.

[13] S. Mei, W. Fei, and S. Zhou, "Gene ontology based transfer learning for protein subcellular localization," *BMC bioinformatics*, vol. 12, no. 1, p. 44, 2011.

[14] Y. Tang, F. Gao, J. Feng, and Y. Liu, "Fingernet: An unified deep network for fingerprint minutiae extraction," *arXiv preprint arXiv:1709.02228*, 2017.

[15] J. Z. Leibo, Q. Liao, F. Anselmi, W. A. Freiwald, and T. Poggio, "View-tolerant face recognition and hebbian learning imply mirror-symmetric neural tuning to head orientation," *Current Biology*, vol. 27, no. 1, pp. 62–67, 2017.

[16] S. Torquato and G. Stell, "Microstructure of two-phase random media. i. the n-point probability functions," *The Journal of Chemical Physics*, vol. 77, no. 4, pp. 2071–2077, 1982.

[17] N. H. Paulson, M. W. Priddy, D. L. McDowell, and S. R. Kalidindi, "Reduced-order structure-property linkages for polycrystalline microstructures based on 2-point statistics," *Acta Materialia*, vol. 129, pp. 428–438, 2017.

[18] M. I. Latypov and S. R. Kalidindi, "Data-driven reduced order models for effective yield strength and partitioning of strain in multiphase materials," *Journal of Computational Physics*, 2017.

TABLE IV
CLASSIFICATION ACCURACY OF EACH SUBSET DATA OF STANDARD CNN NO.1 FOR CRYSTAL PLASTICITY DATASET

Width (μm)	**2**	**1**	**0.5**	**0.25**	**0.125**	**0.0625**
Small-reload & two slips	94.44%	94.44%	97.22%	97.22%	97.22%	97.22%
Small-reload & one slip	100%	100%	100%	100%	100%	83.33%
large-reload & two slips	97.22%	97.22%	100%	94.44%	86.11%	75.00%
large-reload & one slip	**83.33%**	100%	100%	83.33%	50.00%	**66.67%**

TABLE V
CLASSIFICATION ACCURACY OF EACH SUBSET DATA OF HYBRID CNN NO.3 FOR CRYSTAL PLASTICITY DATASET

Width (μm)	**2**	**1**	**0.5**	**0.25**	**0.125**	**0.0625**
Small-reload & two slips	94.44%	94.44%	100%	97.22%	94.44%	100%
Small-reload & one slip	100%	100%	100%	100%	100%	100%
large-reload & two slips	94.44%	97.22%	100%	94.44%	94.44%	**80.56%**
large-reload & one slip	100%	100%	100%	83.33%	**83.33%**	**66.67%**

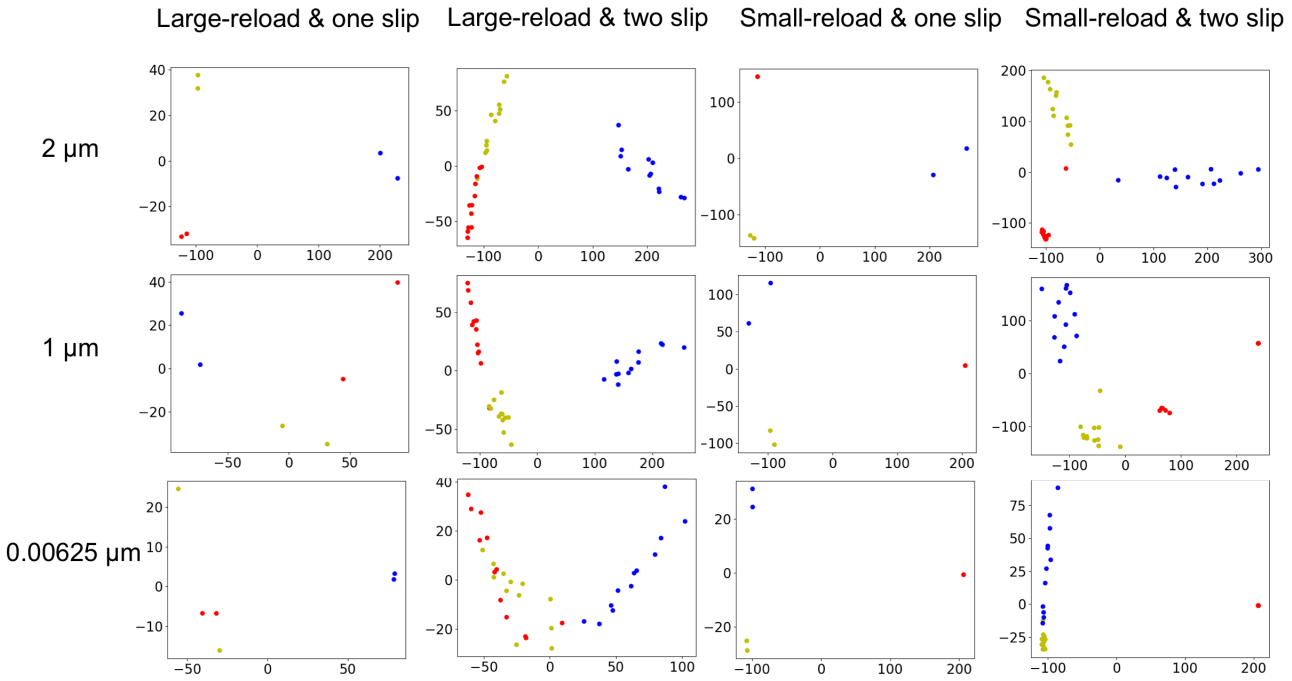

Fig. 6. Projection of each subset data on first two principal components of the outputs of the second fully connected layer of hybrid CNN No.3 for crystal plasticity dataset. Red color, yellow color and blue color denote samples labeled with 0.1%, 1% and 10% strain prior loaded state, respectively.

[19] D. Jeong, L. Dai, M. Kamionkowski, and A. S. Szalay, "The redshift-space galaxy two-point correlation function and baryon acoustic oscillations," *Monthly Notices of the Royal Astronomical Society*, vol. 449, no. 3, pp. 3312–3322, 2015.

[20] Y. Wang, R. Brunner, and J. Dolence, "The sdss galaxy angular two-point correlation function," *Monthly Notices of the Royal Astronomical Society*, vol. 432, no. 3, pp. 1961–1979, 2013.

[21] Z. Yang, Y. C. Yabansu, R. Al-Bahrani, W.-k. Liao, A. N. Choudhary, S. R. Kalidindi, and A. Agrawal, "Deep learning approaches for mining structure-property linkages in high contrast composites from simulation datasets," *Computational Materials Science*, vol. 151, pp. 278–287, 2018.

[22] P. E. Anuta, "Spatial registration of multispectral and multitemporal digital imagery using fast fourier transform techniques," *IEEE transactions on Geoscience Electronics*, vol. 8, no. 4, pp. 353–368, 1970.

[23] S. Papanikolaou, M. Tzimas, H. Song, A. C. Reid, and S. A. Langer, "Learning crystal plasticity using digital image correlation: Examples from discrete dislocation dynamics," *arXiv preprint arXiv:1709.08225*, 2017.

[24] V. Nair and 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.

[25] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, 2010, pp. 249–256.

[26] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[27] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International Conference on Machine Learning*, 2015, pp. 448–456.

[28] M. Lin, Q. Chen, and S. Yan, "Network in network," *arXiv preprint arXiv:1312.4400*, 2013.