Available online at www.sciencedirect.com

ScienceDirect

Procedia Manufacturing 48 (2020) 992–999

Procedia MANUFACTURING

www.elsevier.com/locate/procedia

48th SME North American Manufacturing Research Conference, NAMRC 48 (Cancelled due to COVID-19)

Using Deep Image Colorization to Predict Microstructure-Dependent Strain Fields

Pranav Milind Khanolkar[a], Aaron Abraham[a], Christopher McComb[b], Saurabh Basu[a,*]

[a]*Harold and Inge Marcus Department of Industrial Engineering, The Pennsylvania State University, State College 16802, US*
[b]*School of Engineering Design, Technology and Professional Programs, The Pennsylvania State University, State College 16802, US*

* Corresponding author. Tel.: +0-000-000-0000 ; fax: +0-000-000-0000. *E-mail address:* sxb514@psu.edu

Abstract

The microstructure of a material governs mechanical properties such as strength and toughness. Various finite element analysis (FEA) software packages are used to perform structural analyses such as predicting the flow of strain or strain fields in a microstructure. Engineers frequently operate these software packages to evaluate mechanical behavior and predict failure. Even though these FEA software packages provide highly accurate analyses, they are computationally intensive, taking a significant amount of time to produce a solution. The time required by the FEA software packages to achieve accurate results largely depends on microstructure details and mesh resolution, thus providing a trade-off between fidelity and computation time. This research proposes the use of Deep Learning algorithms to achieve a significant reduction in the time required to predict high-accuracy strain fields in a two-dimensional microstructure with defects. This work presents a foundation for developing deep neural networks to conduct structural analyses, thus reducing the exclusive use of computationally demanding FEA software and augmenting the analytical capabilities of scientists and engineers.

Keywords: Deep Learning; Strain fields; Convolutional Neural Networks; Microstructure

1. Introduction

Structural analysis is an important tool for investigating the integrity of a structure by evaluating its stress and strain response under specific boundary conditions. This is often performed using finite element analysis to pinpoint locations that may experience maximum stress and strain that could lead to failure. In this regard, structural analysis helps to assess whether a given structure is able to withstand required loading conditions adequately during its intended life. Finite Element Analysis (FEA), simulations are performed using software such as ANSYS Workbench, COMSOL Multiphysics, ABAQUS, Autodesk Simulation, and FEA Multiphysics and these software packages also facilitate changes in the design might improve functionality.

Microscale structural analysis can be performed to study the effects of microstructure defects in accelerating failure. Although the aforementioned software packages are able to conduct such analysis with high accuracy, the time taken to predict the results largely depend on the computational power of the device as well as the complexity of the physics governing behavior of the microstructure. As this complexity increases, the time to predict accurate results increases as well. Predicting the performance of extremely complex microstructures becomes burdensome, requiring a substantial amount of time to complete a single prediction/iteration. Therefore, there is a need to develop suitable methods that compute and predict results in a shorter time without significant loss of accuracy. Machine learning algorithms have proven quite useful to predict and classify various entities based on available data in relatively short time and will be used here to accomplish rapid FEA.

Deep learning [1], a subset of machine learning is a powerful tool in feature recognition that has various applications [2] such as handwriting detection [3,4], face

2351-9789
10.1016/j.promfg.2020.05.138

detection or recognition [5,6], and natural language processing [7,8] across different fields of science [9–11], engineering [12] and design [13–15]. Convolutional Neural Networks (CNNs) [16,17] facilitate deep learning that are particularly useful for the analysis of data that can be represented as images. Shea and Nash [17] have clearly depicted the use of CNNs and their uniqueness in recognizing patterns through series of layers, that makes CNNs suitable for image-oriented applications.

Recent research in the field of machine learning has focused on the prediction of material behavior and properties based on microstructural characteristics [18–20]. The current research focusses on the prediction of strain fields due to a uniform displacement boundary condition imposed on a microstructure with defects. Specifically, this work will predict strain fields associated with a two–dimensional (2-D) microstructures with defects. To accomplish this analysis using deep learning, we will test the efficacy of image colorization CNNs [21,22]. CNNs of this type take a greyscale image as input (here, the microstructure), and provide three images on red, green, and blue channels as output. We hypothesize that problem of prediction of strain fields can be treated as an image colorization problem, wherein the microstructure image can be treated as the original gray scale image (e.g. in conventional image colorization) and strain components (e.g. ε_{11}, ε_{22} and ε_{12}) are predicted as output color channels.

This problem is motivated from challenges in advanced manufacturing of complex geometries and microstructures. Volumetric microstructure defects, e.g. porosity, are often produced in these scenarios and must be mitigated before the fabricated component is deployed. However, routes for their mitigation are challenging and require understanding of the behavior of the advanced geometry + microstructure + defect triad under various loading conditions. Occasionally, these defects are too challenging to mitigate and hence, components may be designed and fabricated with the assumption that they will contain tolerable defects, thus also necessitating an understanding of their behavior in complex loading conditions. However, traditional computational routes, e.g. FEA might be too resource intensive to address this requirement.

The remainder of the paper is organized as follows. The background section provides a brief overview of the current research and developments made in the field of material sciences using FEA along with a brief review of the basics of deep learning. The next section describes the methodology of this work including data generation and the application of the deep learning algorithm. In the next section, the CNN is trained and tested on data generated using a Finite Element Analysis software. Several experimental analyses highlight the effect of changing the number and size of defects in the microstructures on accuracy and computation time performance of the CNN (and FEA, as a baseline comparison). The final section ends with discussion of conclusions and future work.

2. Background

Modern advanced materials are characterized by a wide range of mechanical, optical or thermodynamic properties those of which are frequently attributed to a microstructure [23]. Herein, the advent of computational modelling has enabled researchers to conduct various parametric studies on a microstructure and improve coupled processes. Computational modelling is extremely useful in both obtaining specific material properties and also in designing new microstructures [23]. To this end, FEA software is useful for analyzing the properties of novel microstructures.

FEA software has become a very important aspect of industrial-level and academic research. Initially researchers used to study macroscopic behavior of microstructures like chemical properties, crystal structure and crystal orientation by using programs like Object-Oriented Finite (OOF) elements and Portable Pixel Map to OOF format translator (ppm2oof) [24]. Programs like OOF enabled the use of mean-field and other spatial averaging methods to produce approximate models for materials behavior [24]. Before the advent of OOF programs, it was significantly more challenging to correlate microstructures with important properties that affected the integrity of the structure. With the development of such a software, a finite element mesh and an easy-to-use graphical interface became the standard for structural analysis of any component. The purpose of such software is to assign properties to features in a material's image and generate a finite-element mesh representing the material useful for conducting desired analysis.

This work specifically makes use of ABAQUS for FEA simulations. ABAQUS is one widely-used software for performing detailed simulations [25]. In ABAQUS several parameter configurations with different computational costs are utilized for finite element analysis such as stress-strain prediction, flow of temperature etc. [26]. The various types of algorithms present in ABAQUS provides different types of advantages and disadvantages, based on the problem that needs to be solved. Although ABAQUS is a very useful software for finite element analysis especially for choosing the correct algorithm for a problem, researchers often face difficulties when performing analysis since the speed and performance level of the software is limited by the processing power of the hardware used.

FEA software can take a significant amount of time to yield a converging solution, and this time increases substantially when detailed, e.g. high spatial resolution results are necessary or when a complex microstructure is analyzed. These shortcomings of FEA software highlight the need for better analysis tool to provide results with high accuracy in relatively short time. With the increasing applications of machine learning algorithms in the field of material sciences [19,20,27–29], both prediction [30–32] and classification [33–35] of mechanical properties and material parameters can be done more quickly and with less computation power than traditional analyses . Work by others has effectively used neural networks to predict shrinkage in molds [30], analyze properties of composites [31,32], provide classification of steel microstructures [33] and perform microstructure-recognition through classification [34]. The utilization of neural networks provided motivation for the work performed here to predict strain fields in microstructures using CNNs.

Generally, deep learning is used to model and understand that contain complex relationships [1,36]. Deep learning algorithms extract patterns often termed as features in the data,

and through a process of self-learning provide a prediction such that the final error computed between the actual output and the predictions is minimum. Many deep learning algorithms entail the use of artificial neural networks. These are computing systems that can solve complex real-world problems through layers of interconnected nodes or neurons that perform parallel computations [37]. The reader can find additional background in [2].

Image recognition is one of the most popular applications of deep learning algorithms. Manufacturing applications can benefit from utilizing CNNs, where imaging is primarily used for analyzing the material properties through examining the effect of the certain boundary conditions on the given material. Furthermore, works involving surveying three-dimensional structures [12,15] as well as additive manufacturing [13] have successfully incorporated the use of CNNs to achieve their desired results. There are numerous deep neural networks that are used for image recognition, classification and manipulation [34,38–41]. One type of image manipulation involves changing the color scheme of the input image. An extremely common type of image manipulation is colorization [22,42] in which a colored version of a greyscale image is inferred. The research documented in this paper will highlight a colorization approach to predict strain fields in two-dimensional microstructures with specific boundary conditions. The data generated and use of CNN for this purpose are depicted in the next section.

3. Methodology

3.1. Data generation

Synthetic data was generated in the ABAQUS PDE (Python Development Environment). Specifically, a 2-D microstructure was generated with Aluminum (Al6061 – T6), rectangular dimensions, 38.1 mm as length and 6 mm as breadth, for plane-strain simulation of elastic deformation, according to the ASTM E8/E8M standards [43].

For initial training and testing portion, each of the microstructures were modelled with 100 circular pores with radii randomly selected between 100 and 500 µm with their locations within the microstructures set according to uniform distribution. The finite element analysis conducted involved uniform uniaxial displacement of 0.1 mm in the plane-strain mode to obtain strain fields. Three plane-strain fields obtained were ε_{11}, ε_{22} and ε_{12} as displayed in Fig. 1. Traditionally, deep learning algorithms tend to require a substantial amount of data in order to perform well [29]. To that end, a data generation pipeline was developed to produce 1000 samples.

After the samples were generated, a MATLAB code was developed to convert the 2-D microstructures and their respective strain fields into images. The code converted the microstructure model and its resultant strain fields into images with a resolution of 1 pixel equivalent to 0.06 mm, such that the images consist of 101 pixels in the vertical direction and 636 pixels in the horizontal direction. This resolution was chosen to preserve the major features of the microstructure with the minimum number of pixels. The inputs to the CNN were the 2-D defects-ridden microstructures whose pixel values were either 0 (indicating a defect) or 1 (indicating material), depicted in Fig. 2.

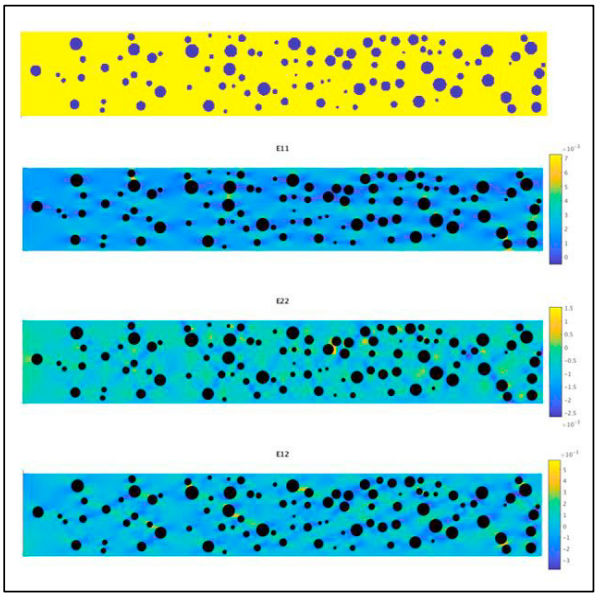

Fig. 1. Microstructure model with porous defects (above); Corresponding strain fields obtained in ABAQUS (below; ε_{11}, ε_{22} and ε_{12} from top to bottom)

The strain fields from the ABAQUS simulations were also converted to images with pixels similar as the input (101 × 636) using the MATLAB code as displayed in Fig 3.

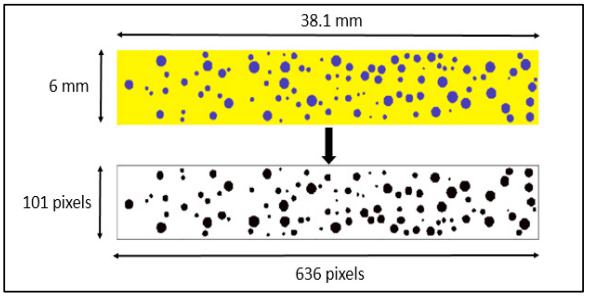

Fig. 2. Microstructure model generated in ABAQUS FEA software (above) and converted to image with 101 × 636 pixels using MATLAB (below)

These samples served as training and testing data for the development of the CNN.

3.2. Proposed Deep Learning Algorithm

A Convolutional Neural Network (CNN) was developed to perform an image colorization as a means of strain field inference. Several CNN architectures, with a varying number of convolutional layers and corresponding parameters such as number of filters, kernel size, activation functions, and strides were evaluated in terms of prediction time and accuracy. In general, increases in the number of convolutional layers, filters sizes and number of filters resulted in increase in time required for training the corresponding CNN and subsequent increase in

time to predict strain fields. Ultimately, an 18-layer CNN consisting 2-D convolutional layers with filters, kernel sizes, padding, activation functions and up-sampling layers, as shown in the Table 1, was developed and selected for this work, based on the accuracy and prediction time of the results.

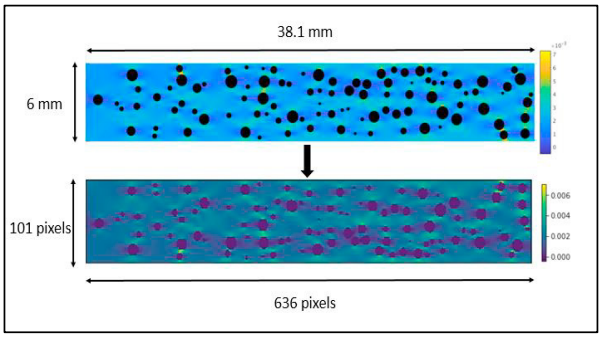

Fig. 3. Microstructure strain field ε_{11} in ABAQUS FEA software (above; displayed using MATLAB) and converted to image with 101 × 636 pixels (below; displayed using Python IDE)

The CNN starts with the input layer defined by the shape obtained in terms of pixels (101×636). The following layers are specified with number of filters to capture maximum detail within the images formed in the layers, kernel size within each layer to set the dimensions of the weights, padding to account for the edges of each images so that all pixels are effectively computed and dimensionality of the images is maintained, appropriate activation functions to provide outputs to every following layer and strides to move the filters across the images to account for every pixel. These layers ultimately converge to the output layer consisting of 3 images each denoting strain fields ε_{11}, ε_{22} and ε_{12} as colored images. Coefficient of determination (R-squared) was used to assess the accuracy of the CNN. The R-squared value was obtained by comparing the predicted output and actual output, both represented by 3 strain field images each.

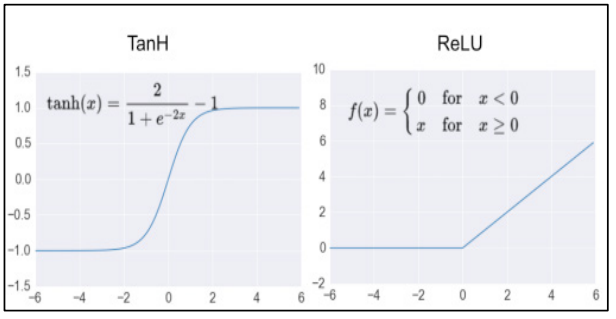

Fig.4. Tanh and ReLU activation functions of CNN[1]

Of the 1000 samples created using the data generation pipeline, 500 samples were used for training and 500 samples were used for testing. This division in datasets is done in order to estimate how the model is expected to perform on samples not used during the training of the model. The CNN was trained for 100 iterations. The Adam optimizer [44] was used for training the CNN with default parameters of 0.001 for learning rate, $\beta_1 = 0.9$ and $\beta_2 = 0.999$. ReLU activation functions were utilized throughout the CNN except for the final two layers where TanH activation function were used to provide appropriate pixel values in accordance to the actual strain values. These activation functions [45] ReLU and TanH are illustrated in Fig. 4.

Table 1. CNN used for Analysis.

Layer No.	Layer Features
0	Input Layer; Shape (101 × 636 × 1)
1	Conv. Layer; Filters: 4 ; Kernel size (9,9); Padding: “same”; Activation: ReLU; Strides: (1,1)
2	Conv. Layer; Filters: 4 ; Kernel size (9,9); Padding: “same”, Activation: ReLU, Strides: (1,1)
3	Conv. Layer; Filters: 8 ; Kernel size (7,7); Padding: “same”, Activation: ReLU, Strides: (1,1)
4	Conv. Layer; Filters: 8 ; Kernel size (7,7); Padding: “same”, Activation: ReLU, Strides: (1,1)
5	Conv. Layer; Filters: 16 ; Kernel size (5,5); Padding: “same”, Activation: ReLU, Strides: (1,1)
6	Conv. Layer; Filters: 16 ; Kernel size (5,5); Padding: “same”, Activation: ReLU, Strides: (1,1)
7	Conv. Layer; Filters: 16 ; Kernel size (3,3); Padding: “same”, Activation: ReLU, Strides: (1,1)
8	Conv. Layer; Filters: 16 ; Kernel size (3,3); Padding: “same”, Activation: ReLU, Strides: (1,1)
9	Conv. Layer; Filters: 16 ; Kernel size (2,2); Padding: “same”, Activation: ReLU, Strides: (1,1)
10	Conv. Layer; Filters: 16 ; Kernel size (2,2); Padding: “same”, Activation: ReLU, Strides: (1,1)
11	Up-Sampling Layer 2-D with size (1,1)
12	Conv. Layer; Filters: 16 ; Kernel size (2,2); Padding: “same”, Activation: ReLU, Strides: (1,1)
13	Up-Sampling Layer 2-D with size (1,1)
14	Conv. Layer; Filters: 8 ; Kernel size (3,3); Padding: “same”, Activation: ReLU, Strides: (1,1)
15	Up-Sampling Layer 2-D with size (1,1)
16	Conv. Layer; Filters: 4 ; Kernel size (7,7); Padding: “same”, Activation: TanH, Strides: (1,1)
17	Up-Sampling Layer 2-D with size (1,1)
18	Conv. Layer; Filters: 3 ; Kernel size (9,9); Padding: “same”, Activation: TanH, Strides: (1,1)

[1] Source: https://www.kdnuggets.com/wp-content/uploads/activation.png

4. Results

A device with GeForce 940M GPU and 6th Gen Intel Core i7-6500U processor as CPU was used for CNN development and simulation. With GPU usage, the trained CNN model provided results with an R-squared values 96.25% obtained from the dataset used for testing. Fig. 5 displays an example of the input data (black and white 2-D microstructure) along with the output data (colored strain fields) obtained from the CNN.

The strain fields which are displayed in Fig 5, were predicted by the CNN with R-squared value of 95.33%. Further analysis was conducted for 50 samples selected randomly from the 1000 samples by comparing the time required by the CNN prediction and the original ABAQUS simulation time. The ABAQUS simulation time considered in this work includes the problem solving time, subsequent to the microstructure modeling and meshing stages. The results of the mentioned computation time comparison are displayed in Fig. 6. The mean computation time required for ABAQUS was found to be 2.23 ± 0.0433 seconds, while the CNN predicted the results in 0.13 ± 0.0171 seconds. This serves as a strong indication that CNNs have efficacy for computing material properties in a rapid fashion that enables quick iteration.

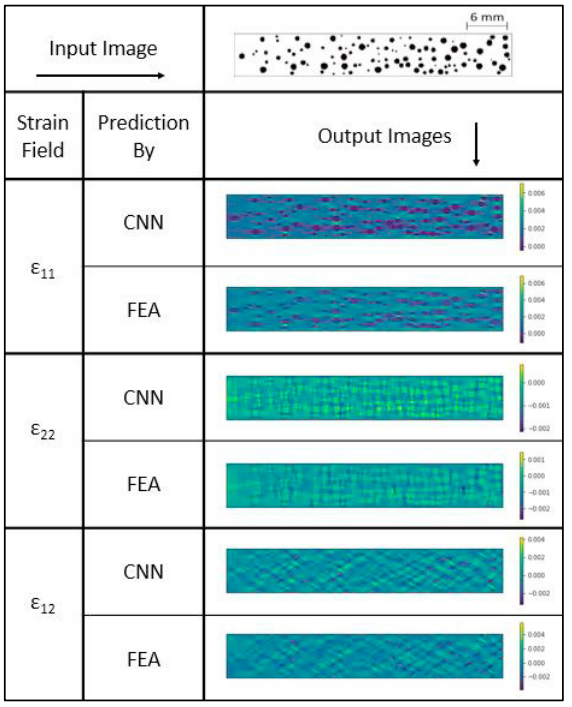

Fig. 5. Microstructure Strain Field predicted by CNN and ABAQUS

Two one-factor experiments were conducted to analyze the effects of varying size and number of defects in microstructures to assess their impact on computation time and accuracy of the CNN in comparison to ABAQUS. In the first experiment, the size of the pores in a microstructure was randomly varied from 10 µm to 500 µm in increments of 50 µm, each range distributed uniformly, with the number of pores kept constant at 50. In the second experiment the size range of the pores was kept constant between 100 – 300 µm and the number of pores incremented from 20 till 200 in uniform increments of 20. For both of these experiments, the original CNN (trained with images of microstructures having 100 pores of size range 100-500 µm) was used to make predictions.

Fig. 7 and 8 present the results of these experiments. The computation time of the CNN and the corresponding accuracy in terms of R-squared values are largely independent of the size and number of the defects. Hence the graph for the prediction times by CNN is always (almost) uniform around 0.1 – 0.2 seconds. In contrast, these values greatly influenced the prediction time exhibited by ABAQUS.

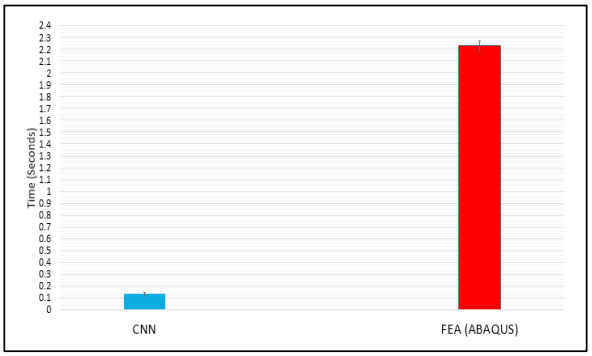

Fig. 6. Comparison of prediction time by CNN and ABAQUS; Error Bars represent Standard Error.

The computation time required by ABAQUS to predict strain fields is displayed in Figs. 7 and 8, and summarized in Table 2 and Table 3, along with the number of elements created during the FEA mesh for the respective microstructures, respectively.

The computation time is dependent on the number of elements created for the model during the meshing as depicted in Table 2 and Table 3. As more defects are generated, or as these defects become smaller, a larger number of elements are required in the mesh, which greatly increases the computational time.

Table 2. Computation time performance of CNN and ABAQUS considering the number of elements formed in the FEA software for a given radii-range of pores within a microstructure.

Radii – Range (µm)	No. of elements formed in ABAQUS	Prediction time by ABAQUS (seconds)	Prediction time by the CNN developed (seconds)
10-50	16188	2.42	0.1097
50-100	17402	2.55	0.1117
100-150	20138	3.01	0.1146
150-200	17953	2.64	0.1116
200-250	14121	2.09	0.1147
250-300	11285	1.66	0.1104
300-350	9156	1.38	0.1127
350-400	7230	1.08	0.1126
400-450	5707	0.87	0.1146
450-500	4663	0.72	0.1117

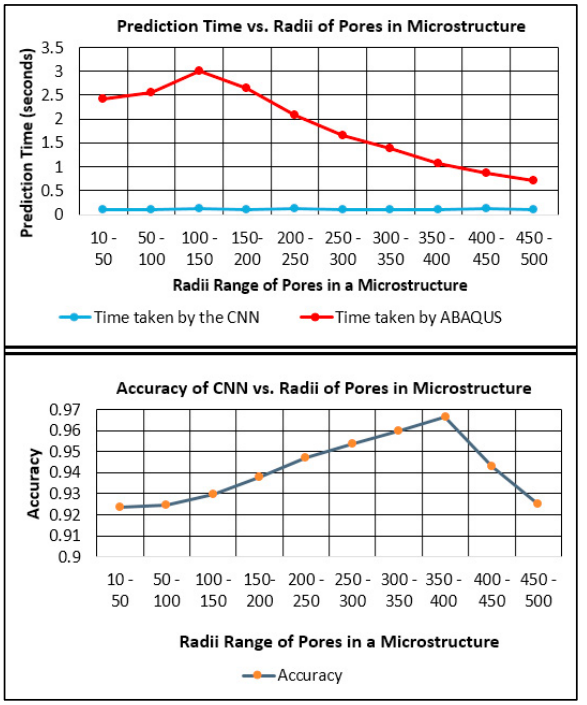

Fig. 7. Comparison of prediction time (above) and accuracy (below) of CNN with ABAQUS by varying Radii-range of microstructures with fixed number of pores

As the radii of the pores increases, the mesh becomes coarser which requires that fewer elements for respective microstructure model for FEA. Precise values for the number of elements in these meshes are tabulated in Table 2. The increase in the radii-range of the pores results in decrease in curvature as well as the decrease in proximity of these pores, which is why the number of elements decrease as the radii range increase. The parabolic shape of FEA prediction times with maximum at 150 µm can be justified based on this reasoning. In contrast, the number of elements in the CNN is constant by definition.

Table 3. Computation time performance of CNN and ABAQUS considering the number of elements formed in the FEA software for a given number of pores within a microstructure.

Number of pores in a microstructure	No. of elements formed in ABAQUS	Prediction time by ABAQUS (seconds)	Prediction time by the CNN developed (seconds)
20	7683	1.24	0.1249
40	11941	1.84	0.1490
60	17123	2.65	0.1196
80	22101	3.35	0.1360
100	29578	4.54	0.1214
120	31682	4.86	0.1196
140	33306	5.22	0.1707
160	30977	4.71	0.1148
180	32717	5.03	0.1256
200	33381	5.14	0.2144

In the case of a fixed radii range (100-300 µm), increase in the number of pores resulted in increase in the number of elements created for the particular samples considered in Fig 8, as displayed in Table 3. Thus, the computation time required the FEA software largely and proportionally depends on the features present in the given model, which in this case are the number of pores and radii-range of pores along with the proximity of these defects/pores that define the number of elements required for meshing.

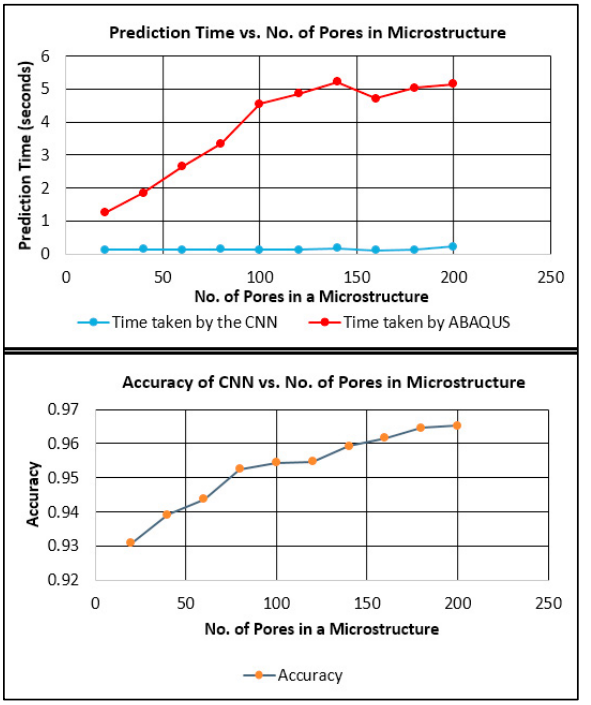

Fig. 8. Comparison of prediction time (above) and accuracy (below) of CNN with ABAQUS by varying number of pores with fixed radii range of 100- 300 µm

Since the weights of the CNN were finalized by training on 500 samples, the computation time required is almost uniform and only dependent on the pixel size of the input image of the microstructure whose strain fields are to be predicted. These weights are set by capturing maximum details and features such as curvature/radii of the pores as well as the proximity of these pores within the trained microstructures. In such manner, the accuracy of predicted strain fields of the microstructures largely depends on the finalized weights set by the training dataset of 500 samples of microstructures each having different defect layout and size. Thus, different microstructures, with distinct range of radii coupled with their locations within the microstructure, will be predicted with different accuracies as depicted in Figs. 7 and 8. As displayed in Figs. 7 and 8, the accuracies of predicted microstructures, with varied radii range and number of pores along with their locations set randomly following a uniform distribution, are obtained within the range 92-97%, which is remarkably high, proving that the CNN works for wide range of defect layout and pore size. Furthermore, strain fields of microstructures with defects in

shape of ellipses were predicted using the CNN. These predictions were made with mean accuracy of 95.20%.

5. Summary and Future work

FEA software, such as ABAQUS, requires a considerable amount of time to predict results of structural analysis. Changes in the complexity of the structural model, such as increasing the number or decreasing the size of microstructural defects, may lead to increase in the computation time. This highlights a need to reduce the computation time in FEA analysis in order to enable more rapid iteration by engineers. This work attempts to meet that need by developing an image-colorization CNN to predict strain fields in microstructures.

The overall accuracy of the CNN developed was approximately 96%. The computation time performance of the CNN developed in predicting the strain fields was almost uniform in the range 0.1-0.2 seconds, irrespective of the shape, number and position of the defects within the microstructure, whereas the computation time required by FEA software fluctuated between 1 to 5 seconds based on the size and number of defects present. Furthermore, it was observed that changes in the number, size and position of the defects in the microstructures had a significant effect on the computation time of the FEA software because these variables resulted in changes to the number of elements created in the FEA mesh. To summarize the main outcome of this project, the time required to predict strain fields by CNN based colorization algorithm was significantly less than the time required by FEA software (ABAQUS), as displayed in Figs. 7, 8 and 9.

There are certain limitations to this research. The CNN developed was trained on synthetic data, generated specifically for this work. Any microstructure with a complex shape and size of defects other than circular pores may not be predicted accurately. Moreover, non-obvious characteristics that are present in empirically-measured data may lead to low prediction accuracy, as CNNs are often not robust to variability in data. Even though the CNN predictions require less time, the training of the CNN requires considerable amount of time and computation power, often meaning a GPU for effective learning. Therefore, it is likely that CNNs such as that developed here will be most useful in applications where a large number of simulations will be performed for a consistent set of input geometries. Otherwise, the large amount of training time required will not be outweighed by a comparable amount of analysis time. In addition to that, this CNN is only capable of predicting strain fields of microstructures with the input image shape, boundary conditions, and defect characteristics on which it was trained. It is important to note that the CNN must be retrained for other microstructure images.

Future work can be done to address these issues by including various shapes of defects and their locations, set through different distributions such as normal, exponential along with clustering these defects, to address a wide range of microstructures. Furthermore, research work using CNNs can be done to predict mechanical behavior within microstructures with non-homogenous material properties, including inclusions of varying sizes and compositions. In addition to that, microstructure strain fields due to different boundary conditions can be labelled based on their boundary conditions and utilized for training the CNN to predict strain fields for microstructures with specified boundary conditions within those labelled ones.

Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant No. 1825686. Any opinions, findings, conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

[1] Yann LeCun, Yoshua Bengio GH. Deep learning (2015), Y. LeCun, Y. Bengio and G. Hinton. Nature 2015.

[2] Deng L. A tutorial survey of architectures, algorithms, and applications for deep learning. APSIPA Trans Signal Inf Process 2014. https://doi.org/10.1017/atsip.2013.9.

[3] Deng L. The MNIST database of handwritten digit images for machine learning research. IEEE Signal Process Mag 2012. https://doi.org/10.1109/MSP.2012.2211477.

[4] Maitra D Sen, Bhattacharya U, Parui SK. CNN based common approach to handwritten character recognition of multiple scripts. Proc. Int. Conf. Doc. Anal. Recognition, ICDAR, 2015. https://doi.org/10.1109/ICDAR.2015.7333916.

[5] Kwolek B. Face detection using convolutional neural networks and gabor filters. Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), 2005. https://doi.org/10.1007/11550822_86.

[6] Li SZ, Lu J. Face recognition using the nearest feature line method. IEEE Trans Neural Networks 1999. https://doi.org/10.1109/72.750575.

[7] Lee H, Yan L, Pham P, Ng AY. Unsupervised feature learning for audio classification using convolutional deep belief networks. Adv. Neural Inf. Process. Syst. 22 - Proc. 2009 Conf., 2009.

[8] Young T, Hazarika D, Poria S, Cambria E. Recent trends in deep learning based natural language processing [Review Article]. IEEE Comput Intell Mag 2018. https://doi.org/10.1109/MCI.2018.2840738.

[9] Goh GB, Hodas NO, Vishnu A. Deep learning for computational chemistry. J Comput Chem 2017. https://doi.org/10.1002/jcc.24764.

[10] Ching T, Himmelstein DS, Beaulieu-Jones BK, Kalinin AA, Do BT, Way GP, et al. Opportunities and obstacles for deep learning in biology and medicine. J R Soc Interface 2018. https://doi.org/10.1098/rsif.2017.0387.

[11] Baldi P, Sadowski P, Whiteson D. Searching for exotic particles in high-energy physics with deep learning. Nat Commun 2014. https://doi.org/10.1038/ncomms5308.

[12] McComb C. Toward the Rapid Design of Engineered Systems Through Deep Neural Networks. Des. Comput. Cogn. '18, 2019. https://doi.org/10.1007/978-3-030-05363-5_1.

[13] Williams G, Meisel NA, Simpson TW, McComb C. Design Repository Effectiveness for 3D Convolutional Neural Networks: Application to Additive Manufacturing. J Mech Des 2019;141:1–12. https://doi.org/10.1115/1.4044199.

[14] Raina A, McComb C, Cagan J. Learning to Design From Humans:

Imitating Human Designers Through Deep Learning. J Mech Des 2019;141:1–11. https://doi.org/10.1115/1.4044256.

[15] McComb C, Murphey C, Meisel N, Simpson TW. Predicting Part Mass, Required Support Material, and Build Time via Autoencoded Voxel Patterns. 29th Annu Int Solid Free Fabr Symp 2018.

[16] Wu J. Introduction to convolutional neural networks. Natl Key Lab Nov Softw Technol 2017. https://doi.org/10.1007/978-3-642-28661-2-5.

[17] Shea KO, Nash R. An Introduction to Convolutional Neural Networks n.d.:1–11.

[18] Butler KT, Davies DW, Cartwright H, Isayev O, Walsh A. Machine learning for molecular and materials science. Nature 2018. https://doi.org/10.1038/s41586-018-0337-2.

[19] Bhadeshia HKDH, Dimitriu RC, Forsik S, Pak JH, Ryu JH. Performance of neural networks in materials science. Mater Sci Technol 2009. https://doi.org/10.1179/174328408X311053.

[20] Bhadeshia HKDH. Neural networks in materials science. ISIJ Int 1999. https://doi.org/10.2355/isijinternational.39.966.

[21] Baldassarre F, Morín DG, Rodés-Guirao L. Deep Koalarization: Image Colorization using CNNs and Inception-ResNet-v2 2017:1–12.

[22] Zhang R, Isola P, Efros AA. Colorful image colorization. Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), 2016. https://doi.org/10.1007/978-3-319-46487-9_40.

[23] Janssens KG, Raabe D, Kozeschnik E, Miodownik MA NB. Computational materials engineering: an introduction to microstructure evolution. Academic Press; 2010.

[24] Langer SA, Fuller ER, Carter WC. Oof: An image-based finite-element analysis of material microstructures. Comput Sci Eng 2001. https://doi.org/10.1109/5992.919261.

[25] Gardner JD. Comparative Study of Finite Element Simulation Software. Consort Deburring Edge Finish 2005.

[26] Sun JS, Lee KH, Lee HP. Comparison of implicit and explicit finite element methods for dynamic problems. J Mater Process Technol 2000. https://doi.org/10.1016/S0924-0136(00)00580-X.

[27] Abbassi F, Belhadj T, Mistou S, Zghal A. Parameter identification of a mechanical ductile damage using Artificial Neural Networks in sheet metal forming. Mater Des 2013. https://doi.org/10.1016/j.matdes.2012.09.032.

[28] Skinner AJ, Broughton JQ. Neural networks in computational materials science: Training algorithms. Model Simul Mater Sci Eng 1995. https://doi.org/10.1088/0965-0393/3/3/006.

[29] Sha W, Edwards KL. The use of artificial neural networks in materials science based research. Mater Des 2007. https://doi.org/10.1016/j.matdes.2007.02.009.

[30] Lee SC, Youn JR. Shrinkage Analysis of Molded Parts Using Neural Network. J Reinf Plast Compos 1999. https://doi.org/10.1177/073168449901800205.

[31] Zhang Z, Friedrich K. Artificial neural networks applied to polymer composites: A review. Compos Sci Technol 2003. https://doi.org/10.1016/S0266-3538(03)00106-4.

[32] Hassan AM, Alrashdan A, Hayajneh MT, Mayyas AT. Prediction of density, porosity and hardness in aluminum-copper-based composite materials using artificial neural network. J Mater Process Technol 2009. https://doi.org/10.1016/j.jmatprotec.2008.02.066.

[33] Azimi SM, Britz D, Engstler M, Fritz M, Mücklich F. Advanced steel microstructural classification by deep learning methods. Sci Rep 2018. https://doi.org/10.1038/s41598-018-20037-5.

[34] Chowdhury A, Kautz E, Yener B, Lewis D. Image driven machine learning methods for microstructure recognition. Comput Mater Sci 2016. https://doi.org/10.1016/j.commatsci.2016.05.034.

[35] Ziletti A, Kumar D, Scheffler M, Ghiringhelli LM. Insightful classification of crystal structures using deep learning. Nat Commun 2018. https://doi.org/10.1038/s41467-018-05169-6.

[36] Arel I, Rose D, Karnowski T. Deep machine learning-A new frontier in artificial intelligence research. IEEE Comput Intell Mag 2010. https://doi.org/10.1109/MCI.2010.938364.

[37] Jain AK, Mao J, Mohiuddin KM. Artificial neural networks: A tutorial. Computer (Long Beach Calif) 1996. https://doi.org/10.1109/2.485891.

[38] Xu Y, Mo T, Feng Q, Zhong P, Lai M, Chang EIC. Deep learning of feature representation with multiple instance learning for medical image analysis. ICASSP, IEEE Int. Conf. Acoust. Speech Signal Process. - Proc., 2014. https://doi.org/10.1109/ICASSP.2014.6853873.

[39] Chan TH, Jia K, Gao S, Lu J, Zeng Z, Ma Y. PCANet: A Simple Deep Learning Baseline for Image Classification? IEEE Trans Image Process 2015. https://doi.org/10.1109/TIP.2015.2475625.

[40] Razzak MI, Naz S, Zaib A. Deep learning for medical image processing: Overview, challenges and the future. Lect. Notes Comput. Vis. Biomech., 2018. https://doi.org/10.1007/978-3-319-65981-7_12.

[41] Wang N, Yeung DY. Learning a deep compact image representation for visual tracking. Adv. Neural Inf. Process. Syst., 2013.

[42] Cheng Z, Yang Q, Sheng B. Deep colorization. Proc. IEEE Int. Conf. Comput. Vis., 2015. https://doi.org/10.1109/ICCV.2015.55.

[43] Standard TO, American A, Standard N. ASTM E8M-13a 2014:1–28. https://doi.org/10.1520/E0008.

[44] Kingma DP, Ba JL. Adam: A method for stochastic optimization. 3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc., 2015.

[45] Ramachandran P, Zoph B, Le Q V. Searching for Activation Functions 2017:1–13.