



## MACHINE LEARNING IN DESIGN, SYNTHESIS, AND CHARACTERIZATION OF COMPOSITE MATERIALS

# Artificial Neural Networks Framework for Detection of Defects in 3D-Printed Fiber Reinforcement Composites

GUAN LIN CHEN,<sup>1</sup> KAUSHIK YANAMANDRA ,<sup>1,2</sup>  
and NIKHIL GUPTA ,<sup>1</sup>

1.—Composite Materials and Mechanics Laboratory, Mechanical and Aerospace Engineering Department, Tandon School of Engineering, New York University, Brooklyn, NY 11201, USA.  
2.—e-mail: vsy212@nyu.edu

One of the major challenges in applying tomography methods for detecting defects in composite materials is the large image datasets generated during imaging, which require significant effort for the detection of damage. Machine-learning (ML) methods require a large training dataset and can be efficient in processing tomography datasets for defect detection. Methods need to be developed for processing images to train the ML algorithms, which is the focus of the present work. An additive manufactured fiber reinforced composite material is imaged using a micro-CT scan to generate an image set for defect detection. The microstructures are processed using the binarized statistical image features (BSIF) method for compression without compromising the desired information about defects. The result shows that the convolutional neural network model has a mean square error of 0.001 in fiber orientation prediction, and a scheme has been developed for defect detection based on the predictions obtained from the ML models.

## INTRODUCTION

Additive manufacturing (AM) is playing an important role in the aeronautical, automotive, and medical fields,<sup>1</sup> by making it possible to fabricate parts of complex geometry, internal structural details, and individually customized designs at low cost and small production runs.<sup>2–4</sup> For AM, a wide range of feed materials is available across the entire material spectrum of polymers, metals, ceramics, and composites.<sup>2,5–7</sup> Parts made of glass and carbon fiber-filled polymer matrix composites (PMCs) are now being 3D printed.<sup>8–10</sup> AM methods have evolved significantly in recent years; however, there are still challenges in obtaining high-quality resolution and surface finish for many applications.<sup>11–13</sup> For example, while fabricating a complex-shaped object using material extrusion methods, which is one of the most commonly used by AM methods,<sup>14</sup> an outline is first printed to more accurately define the contours, and then an infill

pattern is used to deposit material within the contour lines.<sup>15</sup> This kind of space-filling scheme leads to gaps at the end of the deposited lines, and creates porosity in the specimens if the process is not well optimized, which affects the mechanical properties of the printed part.<sup>16</sup> In addition, the curvatures present in the computer-aided design (CAD) models are often reformed with linear segments in file formats such as STL, and can result in the loss of dimensional accuracy in the printed part.<sup>17,18</sup> Such effects cannot be avoided in material extrusion methods and may lead to voids of several tens, even hundreds, of microns in size near curvatures, which can be detected using both destructive and non-destructive imaging methods.

Quality assessment of AM parts is a challenge that is being addressed by a variety of methods, such as automated or computationally assisted non-destructive inspection,<sup>19</sup> to maintain high quality as well as high manufacturing throughput.<sup>20,21</sup> In AM, since a part is printed in several hundred layers, detection of defects may be possible through in situ monitoring. In a way, the lay-up process of composite manufacturing is similar to the layer-by-layer manufacturing of AM, while the machine-learning

(Received January 9, 2021; accepted April 22, 2021;  
published online May 14, 2021)

(ML) methods applied to composite quality assessment can be applicable to the AM process.<sup>22</sup> ML methods are now being applied to a variety of problems in materials science, including fields such as the development of battery materials and the testing of composite materials,<sup>23,24</sup> where enormous amounts of raw data are generated on a daily basis that can be used to train the model for effective decision-making.<sup>25</sup> The artificial neural network (ANN) is a computational ML method, inspired by the architecture of biological neural networks,<sup>26</sup> that has been applied to many problems related to composite materials<sup>27–30</sup> and other materials for defect detection.<sup>31,32</sup>

ML methods are being applied to materials design and characterization fields.<sup>33–35</sup> The large image dataset generated in many materials characterization methods, such as micro-CT ( $\mu$ CT) scan, which is a limitation for manual inspection methods, is an asset for the training of ML methods. In deep learning methods, massive image datasets are used for training the algorithms and improving the quality of the results.<sup>36</sup> It is beneficial to pre-determine the useful features that need to be detected to reduce the training effort and increase the accuracy of detection of the desired features. The location and orientation of the reinforcing fibers governs the mechanical response of composite materials, which can be detected by ML methods.<sup>37</sup> ML methods have also been used on optical images to identify and classify two-dimensional materials.<sup>38</sup> Use of ML methods on large image databases sometimes requires significant signal processing effort to make the search faster or for identification of the features of interest.

In this work, a glass fiber-reinforced PMC filament is used to fabricate specimens by a commercial material extrusion 3D printer. The specimens are imaged using a  $\mu$ CT scanner. The image dataset has been used to train a ML algorithm. Previous studies have revealed that 2D images with irregular outlines increase the difficulty in analyzing the microstructure features of composite materials, such as the fiber orientation identification,<sup>38,39</sup> and such limitations are overcome by cropping the training images into a circular shape during training. Furthermore, with the help of the binarized statistical image features (BSIF) algorithm, the desired features of the images are retained while reducing the image data to 1D information, and achieving a significant reduction in the size of the dataset. Application of these image-processing methods to reduce the size of the dataset helped in making the training significantly faster. In addition, two different ANN architectures have been used to compare their ability to address such problems. The short fibers present in the 3D printing feed filament are largely aligned in the printing direction. The misalignment of fibers and the gaps between the deposited lines lead to voids

that are identified as the defect. The combined effect of all the voids is detected as the defects in the composite material.

## MATERIALS AND METHODS

### CAD Model Design and 3D Printing

The details of specimen design and AM process have been presented in a previous publication,<sup>39</sup> and are included here only in brief. The sample was designed using SolidWorks 2017 (Dassault Systèmes, Waltham, MA, USA) and saved in STL format. Figure 1 shows the SolidWorks CAD model of the sample and the  $\mu$ CT scan-reconstructed 3D model.<sup>39</sup> The material used for printing was ABS-GF10 glass fiber-reinforced acrylonitrile butadiene styrene (ABS) filaments of 1.75 mm diameter, manufactured by 3DXTECH (Grand Rapids, MI, USA). A FlashForge Creator Pro Dual Extruder 3D printer was used to print the cube-shaped specimens of 6 mm sides. The CAD files were saved in STL format and then sliced using ReplicatorG, and the G-code was generated for printing the model. The printing parameters include 100% solid infill, travel rate of 15 mm/s, feed rate of 41 mm/s, extrusion temperature of 220°C, and the build platform temperature of 120°C. The printing direction of the fiber was set to 0° and 90° for alternate layers, and the layer resolution (thickness of each layer) was set to 0.27 mm. The existence of sharp edges in the cube geometry is a challenge for the 3D printing method, and reversal in the printing direction may generate some porosity in the specimen, which will be identified in this exploratory study.

The printed specimen was mounted on a Skyscan 1127  $\mu$ CT scanner and scanned at an X-ray voltage of 44 kV with 222 uA, and the image pixel size was 9.5  $\mu$ m, the rotation step was 0.6° per scan, with 360° rotation, taking 34:54 min for the scan.

### Micro-CT Scan and Image Reconstruction

The  $\mu$ CT scan images underwent reconstruction using the software NRecon. In this step, the reconstructed image can be tuned with the “smooth” function to adjust the resolution, the “contract” function to reduce noise, and the “rotation” function to rotate the images. Since, the sample was randomly mounted on the stage, the images were rotated to designate 0° as the horizontal direction after the image reconstruction. The glass fibers filled in ABS can be observed as the white lines in the images. The image dataset contains 656 images in jpg format of 683 × 674 pixel size each.

### Dataset Preparation for Machine Learning

Each image slice represents a variety of fiber directions based on the space-filling algorithm used for determining the print head movement. The potential defects in the specimen include fiber

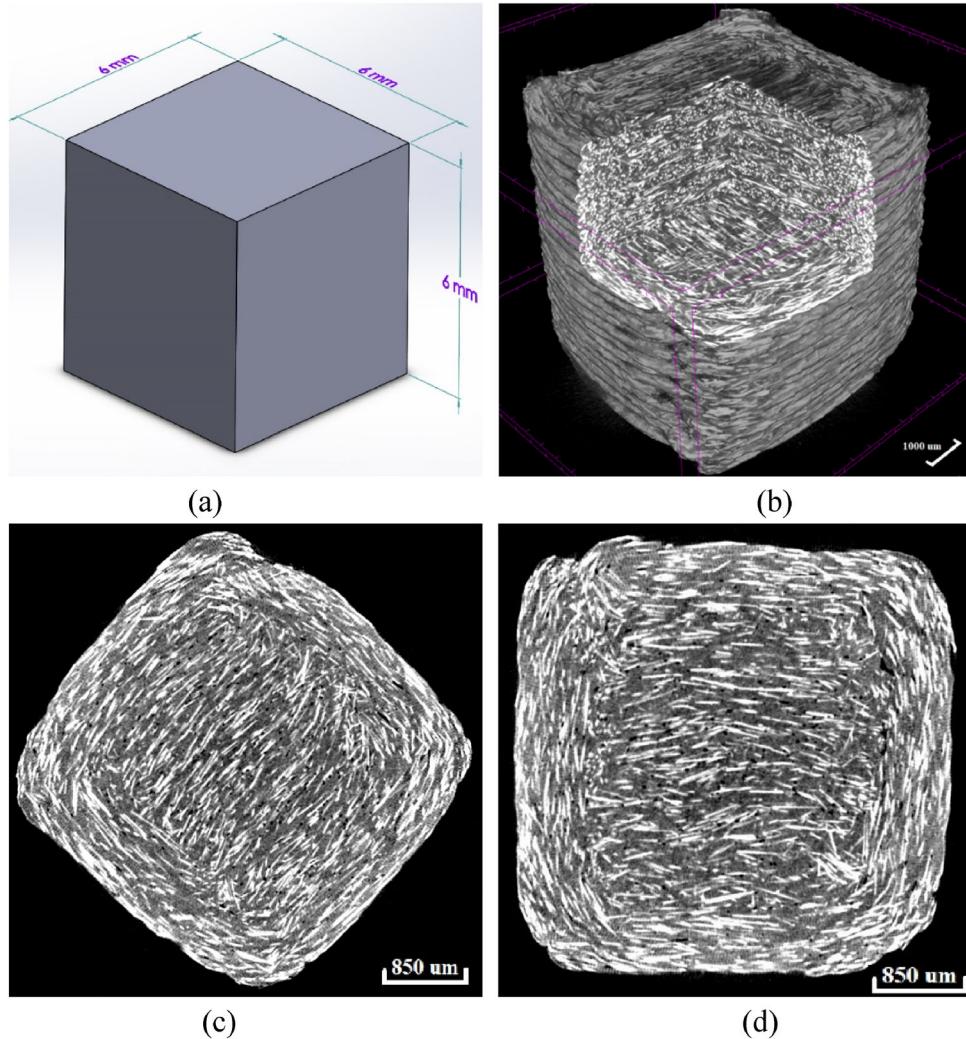


Fig 1. (a) CAD model of a cube with 6 mm side and (b) a  $\mu$ CT scan of the 3D-printed part showing an in situ signatures representation (the violet lines seen in the image are from the imaging reference frame). Images obtained after  $\mu$ CT scan and image reconstruction: (c) a random orientation image, and (d) adjusted  $0^\circ$  rotated image.

direction misalignment and voids. The ML model tuning dataset needs to identify these features, which are significantly smaller than each image. Hence, each image was further sliced in circular sub-images of  $100 \times 100$  pixels using a Matlab code. The sub-sampling process and the resulting images are shown as Fig. 2. This process results in 961 cropped circular images for each  $\mu$ CT image slice. The circular shape helps in reducing the errors caused by the irregular outline of the specimen. In order to train the model, 126 images with clear visibility of the  $0^\circ$  fiber angle were selected, and then the images were synthetically rotated counterclockwise at intervals of  $1^\circ$  each from  $0$ – $180^\circ$  using a Python code to train the model for predicting various rotation angles. The process allows the creating of a dataset with controlled orientation of the glass fiber, resulting in 22,806 images of  $100 \times 100$  pixel size each.

### Binarized Statistical Image Features (BSIF) Process

Since 22,806 images represent a large dataset, the training time for the neural network is reduced by applying the BSIF algorithm, which can significantly reduce the storage size of each image without losing the accuracy for interpreting the features present in the image. BSIF is described as computing a binary code string for the pixels of a given image. The code value of a pixel is considered as a local descriptor of the image intensity pattern in the pixel's surroundings. Histograms of the pixels' code values allow the characterizing of the texture properties within image sub-regions.<sup>40</sup> Thus, the descriptor can be used in texture recognition tasks. An example of the images processed by BSIF is given in Fig. 3. The 2D images are converted into 1D data with 1 row and 256 columns, and the resulting data were saved as a csv file with 22,806 rows of image features. This dataset was divided into

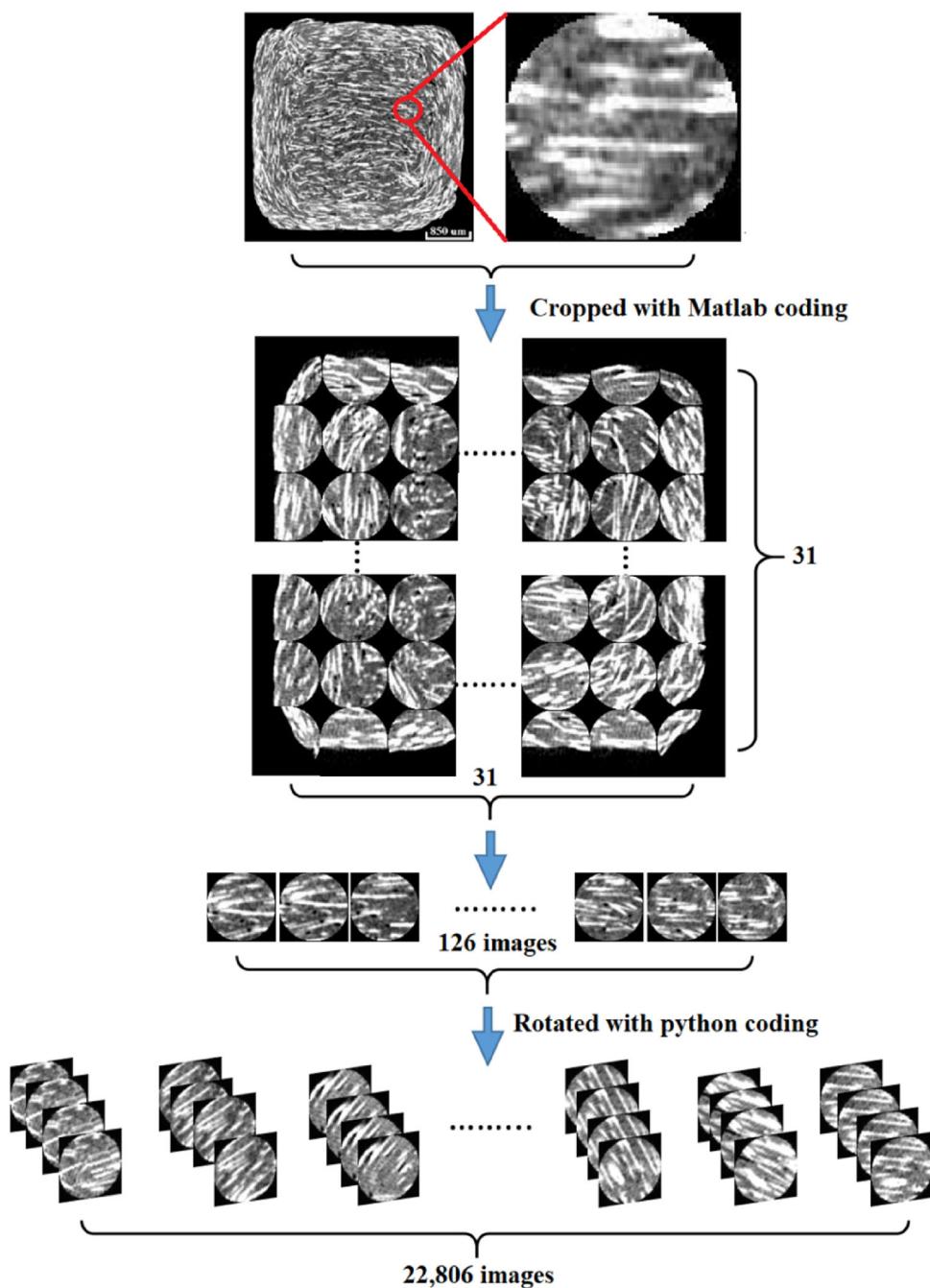


Fig 2.. The image dataset (22,806 segmented images) obtained from the  $\mu\text{CT}$  scan after removing overlapping images and being labeled as  $0^\circ$  for ML.

training, validating, and testing datasets. Since 126 images for each angle are available, and to avoid bias while training the model, 100 images were picked for each angle as the training dataset, which totaled 18,100 sub-images. Then, 20 images from each angle were added to the validating dataset, which totaled 3,620 sub-images. Finally, 1086 sub-images, corresponding to 6 images for each angle, were used in the test dataset. By feeding the dataset

with evenly distributed weights, the neural network will have the least bias when the training is completed.

### Machine Learning

Two kinds of ANN architectures have been used to build  ${}^0\text{models}$  using Python with Tensorflow, A recurrent neural network (RNN) and A

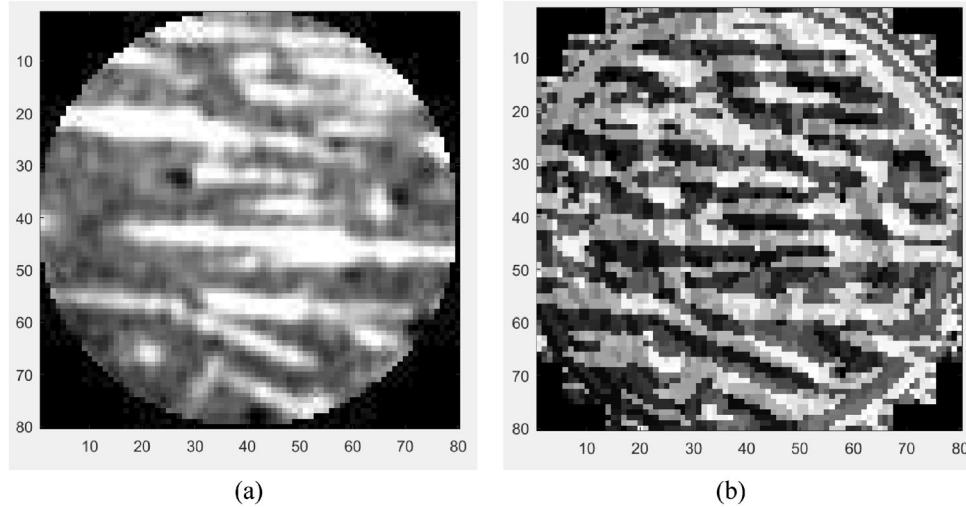


Fig 3. BSIF representation of the glass fiber orientation with a  $100 \times 100$  pixel image: (a) the original circular cropped 2D image, and (b) the 2D image converted by BSIF.

convolutional neural network (CNN). Both neural networks can deal with sequences of variable lengths. The RNN uses the memory-state to process the input data on each neuron. In this work, an RNN architecture with 64 long short-term memory (LSTM) cells has been used to train the model. The CNN model was trained with kernel size 5 and filter size 80 in the first layer, and same parameters were used in the second layer to iterate through the data to train the model.

The BSIF-converted csv file was loaded and converted to an array as (18,100, 1, 256) for the training feature, which means that this training set has 18,100 data, representing the true angle label, with time-step as 1, and 256 feature points. The same process was applied to the validation data, which resulted in an array of (3620, 1, 256). The loss function depends on the desired result parameter, such as the fiber angle. Hence, the mean square error (MSE) was used as the loss function in the model. The estimated values and the actual value of the angle are used to calculate the MSE.

### The Damage Scales

A damage scale has been adopted to quantify the damage detected in the composite through the image analysis. It can be used to compare two specimens analyzed under the same set of conditions. The damage scale,  $D$ , is calculated by:

$$D = \frac{N_d}{N_t} \times 100\% \quad (1)$$

where  $N_d$  is the number of deviated images and  $N_t$  is the number of the total cropped images. The calculation of  $D$  depends on several factors, including the threshold used for calculating the deviation of a fiber from the assigned angle, the size of the

segmented image, the total number of segmented images, and the resolution of the imaging method. The possible deviations are based on factors such as fiber misalignment, fiber debonding, and voids.

## RESULTS AND DISCUSSION

### Prediction of Fiber Orientation

As mentioned in Sect. 2.4, the image dataset has been split into the training set and the validating set with 18,100 and 3620 images, respectively. Since the built-in function of the training set splits the dataset randomly, it preferentially takes the first few data for the neural network to train. This may lead to an uneven weightage based on the trends in the first few data points and so cause bias while predicting. However, with the evenly distributed training and validating data, the possible bias during prediction can be eliminated. To test the accuracy of the trained model, the remaining 1086 images were used to check the prediction accuracy of the model. With the MSE loss function, a continuous output number represents the angle of the fiber that will be predicted. Although the ML algorithm mostly depends on the parameters within each layer of the neural network, with the same set-up, a slight deviation occurred. Therefore, the standard deviation is considered in this research. Hence, 5 outputs for each iteration were saved. Here, in combining the fine-tuning technique mentioned above, an RNN model with MSE 0.87 and a CNN model with MSE 0.001 were achieved. The testing results are shown in Fig. 4. To illustrate the testing results for different epochs, the standard deviation of the 5 training results is shown in Table I.

Figure 4 shows the prediction results of the RNN and CNN models, including the average MSE

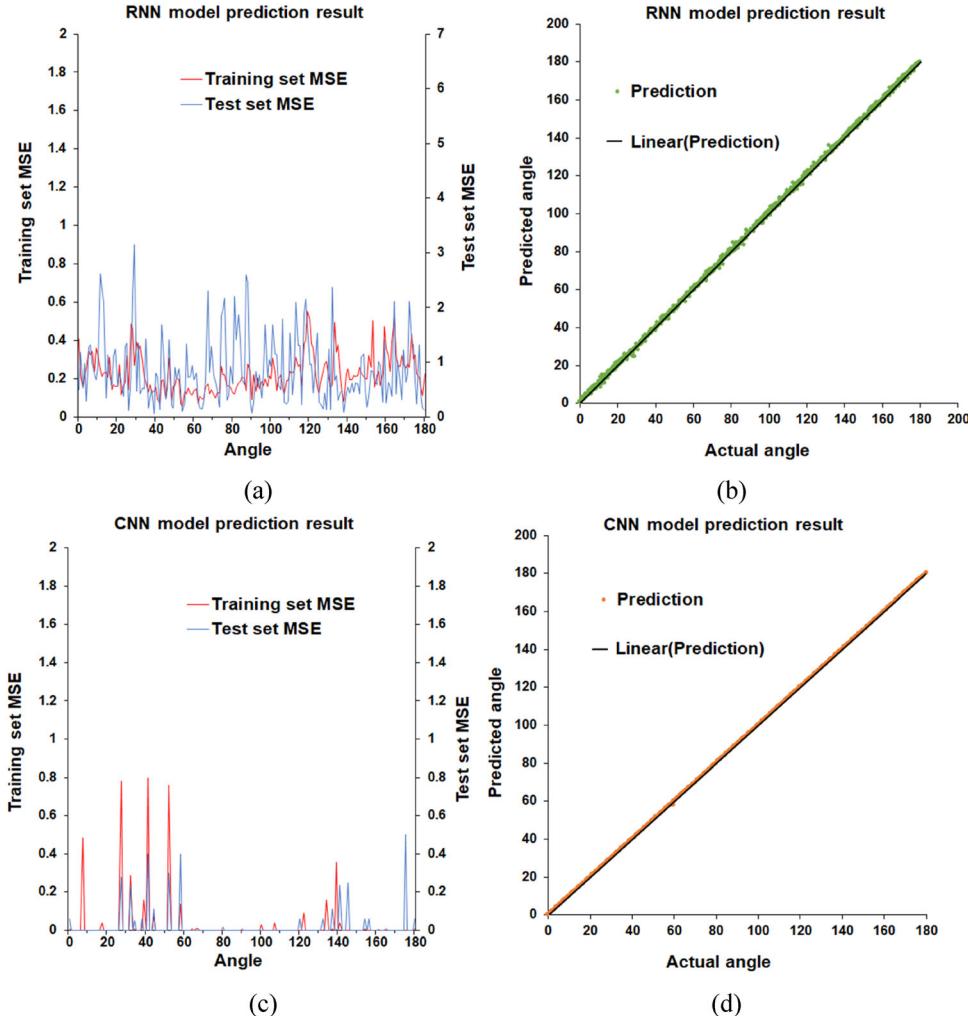


Fig 4. RNN predictions for each angle in the training and test datasets on 10,000 epochs: (a) mean square error and (b) predicted angle of fiber orientation compared to the correct labeled value for the test dataset with the regression line. CNN predictions for each angle in the training and test dataset on 100 epochs: (c) mean square error and (d) predicted angle of fiber orientation compared to the correct labeled value for test dataset with the regression line.

values for each angle in the training and testing datasets in each epoch. The test results show the accuracy over the training dataset. The difference in the MSE values between the training and test dataset is very small. No trend of underfitting or overfitting is observed in the prediction result. The information of the prediction of the fiber angle with respect to the actual angle for the test dataset, and the predicted value for each labeled angle and the linear regression line, are also shown in Fig. 4a, providing the proof that the predicted result is close to the exact angle. Therefore, the trained model has been acquired. Moreover, from the results listed in Table I, the CNN model provides an overall better prediction result compared to the RNN model, and so do the MSE values in the training, validation, and test datasets. Therefore, the following defects detection is conducted with the CNN model.

## Defects detection

The  $\mu$ CT scan image presented in Fig. 5a shows that the 3D printing process first prints an outline of the layer and then fills the space inside the outline at  $0^\circ$  and  $90^\circ$  fiber orientation. This area is used to detect the glass-fiber orientation.

Figure 4c and d and Table I show that the predictions conducted by the trained CNN model have a higher accuracy compared to the RNN model. With the highly accurate machine-learning model, the  $\mu$ CT scan images of the specimen can be used for defect detection. Figure 5 shows a  $\mu$ CT scan slice of the sample. The image was then cropped into small square images to perform a regional glass-fiber orientation prediction. Each square image was cropped using a python code with  $100 \times 100$  pixels from the original image of  $683 \times 674$  pixels. An orientation indicator was generated by the trained machine-learning model to display the directions of the glass fibers in each sub-sampled image, as

**Table I. MSE values at different epochs for training, validation and test datasets in RNN and CNN model predictions.****Standard deviation for RNN Model**

| Training | Dataset        | Epochs |       |       | Standard deviation for 10,000 epochs |
|----------|----------------|--------|-------|-------|--------------------------------------|
|          |                | 500    | 1000  | 10000 |                                      |
| 1        | Training MSE   | 1.551  | 0.838 | 0.312 | Training MSE 0.0074                  |
|          | Validation MSE | 1.950  | 1.469 | 1.272 |                                      |
|          | Test MSE       | 1.894  | 1.539 | 0.983 |                                      |
| 2        | Training MSE   | 1.236  | 0.788 | 0.313 | Validation MSE 0.0128                |
|          | Validation MSE | 1.579  | 1.479 | 1.313 |                                      |
|          | Test MSE       | 1.677  | 1.545 | 1.048 |                                      |
| 3        | Training MSE   | 1.853  | 0.846 | 0.320 | Test MSE 0.0154                      |
|          | Validation MSE | 2.027  | 1.536 | 1.26  |                                      |
|          | Test MSE       | 1.906  | 1.516 | 0.968 |                                      |
| 4        | Training MSE   | 1.243  | 0.795 | 0.389 |                                      |
|          | Validation MSE | 1.864  | 1.379 | 1.314 |                                      |
|          | Test MSE       | 1.525  | 1.439 | 1.010 |                                      |
| 5        | Training MSE   | 1.970  | 0.828 | 0.217 |                                      |
|          | Validation MSE | 2.490  | 1.459 | 1.257 |                                      |
|          | Test MSE       | 2.574  | 1.556 | 1.021 |                                      |

**Standard deviation for CNN Model**

| Training | Dataset        | Epochs |       |        | Standard deviation for 150 epochs |
|----------|----------------|--------|-------|--------|-----------------------------------|
|          |                | 5      | 10    | 150    |                                   |
| 1        | Training MSE   | 0.288  | 0.266 | 0.120  | Training MSE 0.01494              |
|          | Validation MSE | 0.058  | 0.096 | 0.010  |                                   |
|          | Test MSE       | 0.003  | 0.002 | 0.001  |                                   |
| 2        | Training MSE   | 0.244  | 0.239 | 0.112  | Validation MSE 0.00536            |
|          | Validation MSE | 0.064  | 0.099 | 0.0099 |                                   |
|          | Test MSE       | 0.003  | 0.002 | 0.0018 |                                   |
| 3        | Training MSE   | 0.292  | 0.291 | 0.141  | Test MSE 0.00036                  |
|          | Validation MSE | 0.071  | 0.089 | 0.019  |                                   |
|          | Test MSE       | 0.005  | 0.004 | 0.0013 |                                   |
| 4        | Training MSE   | 0.288  | 0.245 | 0.0123 |                                   |
|          | Validation MSE | 0.058  | 0.092 | 0.018  |                                   |
|          | Test MSE       | 0.003  | 0.002 | 0.0012 |                                   |
| 5        | Training MSE   | 0.288  | 0.210 | 0.140  |                                   |
|          | Validation MSE | 0.058  | 0.095 | 0.0199 |                                   |
|          | Test MSE       | 0.003  | 0.002 | 0.0015 |                                   |

shown in Fig. 6a. The position of the indicator in each sub-image represents the local glass-fiber orientation, which provides a full picture of the toolpath of the printed specimen.

Based on the collection of the orientation indicators in Fig. 6b, the defects are observed in places where discontinuous predictions are shown. To properly address the position of the cropped images, a rows and columns index has been added onto the image. Also, with the help of the ML model, 47 deviations were captured. The regions of the deviations observed from the orientation indicators are superimposed on the microstructure of one layer of composite in Fig. 6a, with the locations of the deviations marked by red circles. The trained

algorithm does not directly detect voids in this work. Rather, fiber misalignment is taken as the measure of the defects in the material. Voids appear as black regions in the images and the appearance of more than expected black color in the image is detected as a deviation in the predictions and labeled as a defect. Due to the contrast difference, the primary defects detected in the images are the voids. It is also noted that the efficient detection of voids depends on the image resolution and the dimensions of the segmented images, since each segmented image is reduced to a single indicator of the fiber direction in the predictions from this method.

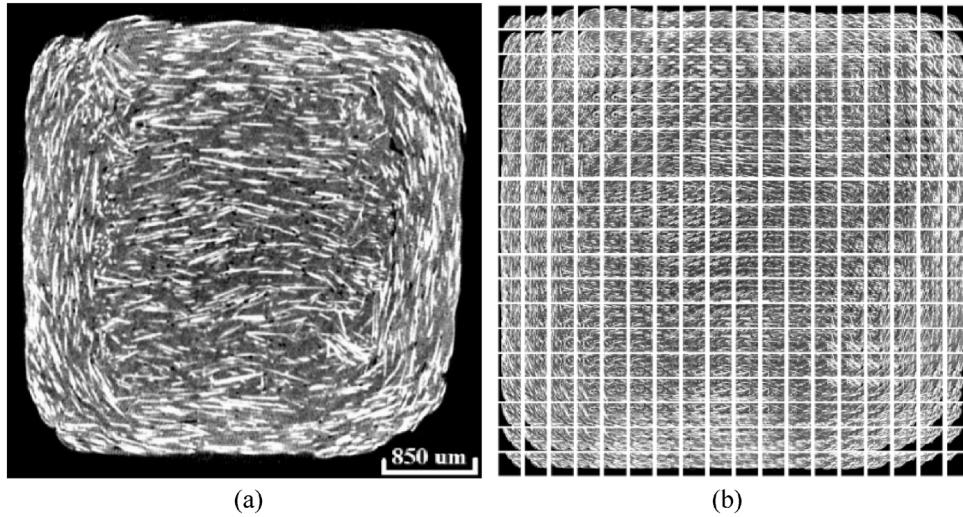


Fig 5. Slice of  $\mu$ CT scan image cropped into small square shapes for regional glass-fiber orientation prediction with (a) the original image and (b) cropped by python coding with 19 rows and 18 columns.

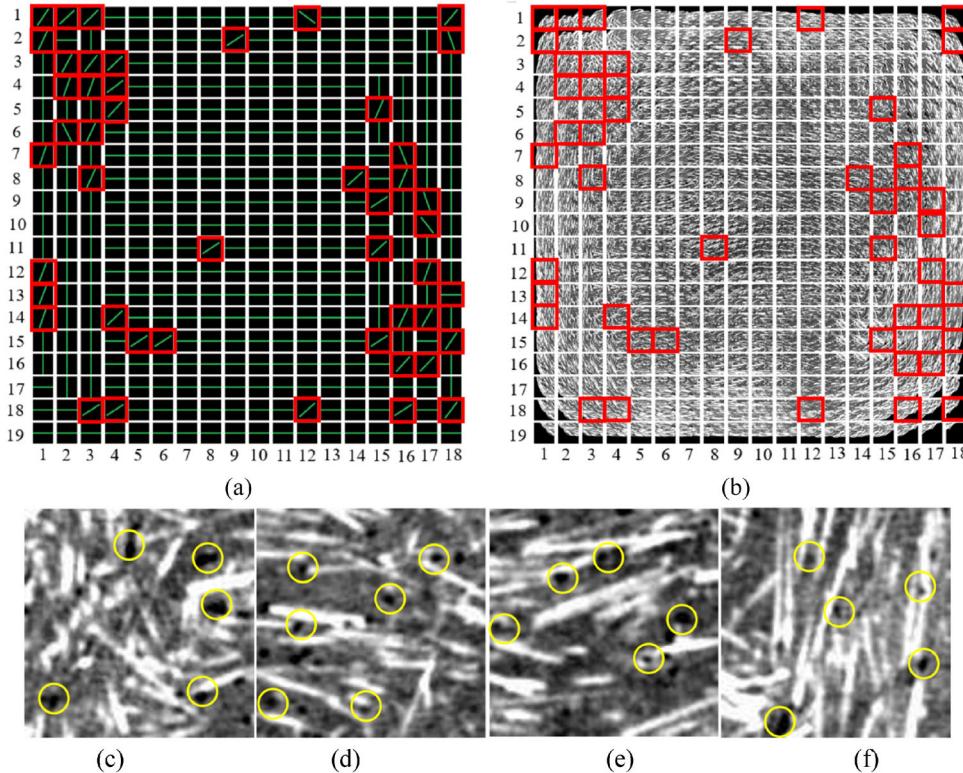


Fig 6. (a) Predicted collection of direction indicators corresponding with (b) cropped image, (c) rows 5, columns 4; (d) rows 8, columns 14, (e) rows 11, columns 8, and (f) rows 14, columns 16, showing the defects marked by yellow circles (Color figure online).

Based on the images showing deviations, a damage scale of 13.7% has been calculated. It should be noted that the deviation can be defined by a number of parameters, including the angle of marker misalignment that is acceptable and should not be counted as the deviation. The porosity present in the specimen results in localized changes in the fiber direction, and a sufficiently small subsampled image is able to detect such aberrations. Examples

of some of the images that are labeled as deviations are magnified as Fig. 6c-f, which shows the defects that resulted in deviations of the predictions. Such defects are marked with yellow circles in the images. Depending on the desired outcomes, the images may be further sub-sampled to a smaller size to more accurately determine the defects. However, the workflow for such detection would be the same.

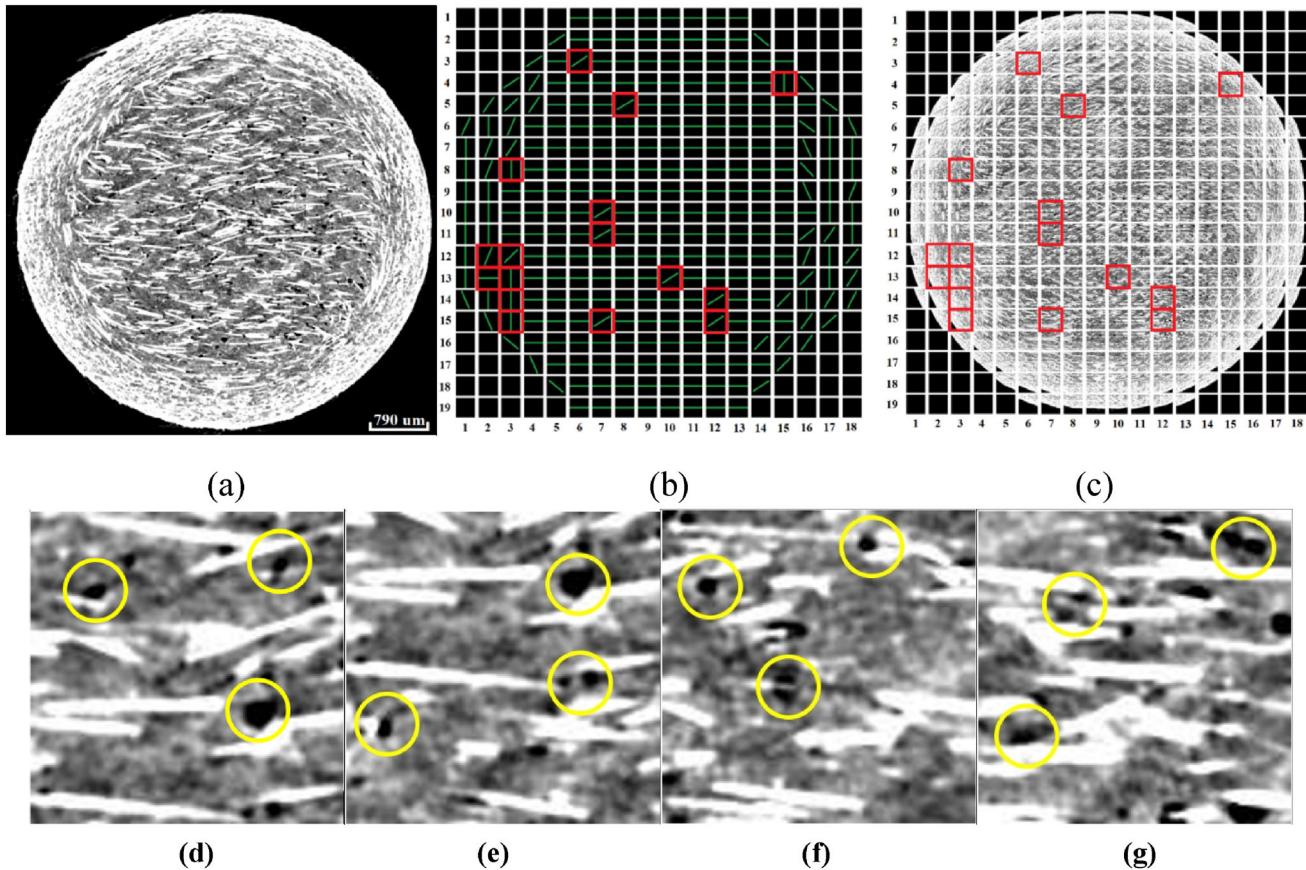


Fig 7. (a) A slice of a micro-CT scan image of the cylindrical sample. (b) A collection of predicted direction indicators corresponding to the micro-CT scan image slice. (c) The cropped micro-CT scan image slice used for prediction. Defects in the image are shown in (d) rows 10, columns 7, (e) rows 11 columns 7, (f) rows 13, columns 10, and (g) rows 14, columns 12, showing the defects marked by yellow circles (Color figure online).

## Validation of the Machine-Learning Model

To further validate the efficiency of the trained CNN ML model, it has been tested on the  $\mu$ CT image set obtained from a specimen of a different geometry. A cylindrical sample, shown in Fig. 7, has been 3D printed using the same material and process as the cubic model used in the previous sections for model training. The images obtained of this specimen have been used to demonstrate the efficiency of the trained algorithm in defect detection. From the predicted result, shown in Fig. 7b and c, the damage areas can be detected. This test on a different specimen demonstrates that the trained ML algorithm can be used effectively for defect detection in composite materials.

## CONCLUSION

This work focuses on training machine-learning (ML) base models for  $\mu$ CT scan images for possible defect detection. The deviation in the fiber direction from the predicted values is considered as the measure of defects in the microstructure. The

possible reasons for deviations in the microstructure include fiber misalignment, voids, and debonds. A glass fiber-reinforced ABS filament was used to 3D print the specimens using a fused filament fabrication 3D printer. The results are summarized as:

- A defect detection method was developed based on artificial neural network architecture applied to  $\mu$ CT scan images of 3D printed specimens of fiber-reinforced composites.
- A refined CNN model achieved an accuracy of a mean square error of 0.001 in predicting the fiber orientation. The model showed that 13.7% of the sub-sampled images had defects in them.
- The trained deep learning model can be used on specimens of a similar nature with a high level of accuracy in the detection of defects.

Although the method has been developed on images acquired after 3D printing, a similar scheme can be used for in situ defect detection during specimens printed using optical images acquired by a camera.

## ACKNOWLEDGEMENTS

National Science Foundation SaTC-EDU grant # 1931724 is acknowledged for supporting this work. The authors thank NYU Tandon School of Engineering Makerspace for the facilities provided for micro-CT scan.

## COMPLIANCE WITH ETHICAL STANDARDS

### CONFLICT OF INTEREST

The authors declare that they have no conflict of interest.

### REFERENCES

1. F.R. Kaschel, R.K. Vijayaraghavan, A. Shmeliow, E.K. McCarthy, M. Canavan, P. McNally, D.P. Dowling, V. Nicolosi, and M. Celikin, *Acta Mater.* 188, 720. (2020).
2. T.D. Ngo, A. Kashani, G. Imbalzano, K.T.Q. Nguyen, and D. Hui, *Compos. Part B* 143, 172. (2018).
3. E.M. Palmero, D. Casaleiz, J. de Vicente, J. Hernández-Vicen, S. López-Vidal, E. Ramiro, and A. Bollero, *Compos. Part A* 124, 105497. (2019).
4. Z. Liu, M. Li, Y. Weng, Y. Qian, T.N. Wong, and M.J. Tan, *Compos. Part B* 193, 108018. (2020).
5. J. Justo, L. Távara, L. García-Guzmán, and F. París, *Compos. Struct.* 185, 537. (2018).
6. A. Averardi, C. Cola, S.E. Zeltmann, and N. Gupta, *Mater. Today Commun.* 24, 100964. (2020).
7. A.K. Singh, B. Patil, N. Hoffmann, B. Saltonstall, M. Doddamani, and N. Gupta, *JOM* 70, 303. (2018).
8. X. Wang, M. Jiang, Z. Zhou, J. Gou, and D. Hui, *Compos. B* 110, 442. (2017).
9. M. Heidari-Rarani, M. Rafiee-Afarani, and A.M. Zahedi, *Compos. Part B* 175, 107147. (2019).
10. A.K. Singh, A.J. Deptula, R. Anawal, M. Doddamani, and N. Gupta, *JOM* 71, 1520. (2019).
11. J.L. Bartlett, A. Jarama, J. Jones, and X. Li, *Mater. Sci. Eng. A* 794, 140002. (2020).
12. Z. Huang, J.-Y. Dantan, A. Etienne, M. Rivette, and N. Bonnet, *Rapid Prototyp. J.* 24, 1524. (2018).
13. H. Kyogoku, and T.-T. Ikeshoji, *Mech. Eng. Rev.* 7, 19–00182. (2020).
14. C. Kim, D. Espalin, A. Cuaron, M.A. Perez, E. MacDonald, and R.B. Wicker, *IEEE Trans. Compon. Packag. Manuf. Technol.* 8, 1469. (2018).
15. T. Kuipers, E.L. Doubrovski, J. Wu, and C.C.L. Wang, *Comput-Aid. Design* 128, 102907. (2020).
16. A. du Plessis, I. Yadroitsava, and I. Yadroitsev, *Mater. Design* 187, 108385. (2020).
17. R. Comminal, M.P. Serdeczny, D.B. Pedersen, and J. Spangenberg, *Add. Manuf.* 29, 100753. (2019).
18. F. Chen, G. Mac, and N. Gupta, *Mater. Design* 128, 182. (2017).
19. F. Honarvar, and A. Varvani-Farahani, *Ultrasonics* 108, 106227. (2020).
20. A.M. Deshpande, A.A. Minai, and M. Kumar, *Procedia Manuf.* 48, 1064. (2020).
21. A. Caggiano, J. Zhang, V. Alfieri, F. Caiazzo, R. Gao, and R. Teti, *CIRP Ann.* 68, 451. (2019).
22. C. Sacco, A.B. Radwan, A. Anderson, R. Harik, and E. Gregory, *Compos. Struct.* 250, 112514. (2020).
23. Y. Liu, B. Guo, X. Zou, Y. Li, and S. Shi, *Energy Storage Mater.* 31, 434. (2020).
24. A. Khan, D.-K. Ko, S.C. Lim, and H.S. Kim, *Compos. Part B* 161, 586. (2019).
25. A. Dogan, and D. Birant, *Exp. Syst. Appl.* 166, 114060. (2021).
26. D.T.T. Do, D. Lee, and J. Lee, *Compos. Part B* 159, 300. (2019).
27. Xu. Xianbo, and N. Gupta, *Adv. Theory Simul.* 2, 1800131. (2019).
28. H. El Kadi, *Compos. Struct.* 73, 1. (2006).
29. A. Sharma, and V. Kushvaha, *Eng. Fract. Mech.* 239, 107328. (2020).
30. O. Kwon, H.G. Kim, M.J. Ham, W. Kim, G.-H. Kim, J.-H. Cho, N.I.I. Kim, and K. Kim, *J. Intell. Manuf.* 31, 375. (2020).
31. L. Tian, Y. Fan, L. Li, and N. Mousseau, *Scr. Mater.* 186, 185. (2020).
32. H. Ko, P. Witherell, Y. Lu, S. Kim, and D.W. Rosen, *Add. Manuf.* 20, 101620. (2020).
33. X. Xu, M. Elgamal, M. Doddamani, and N. Gupta, *J. Compos. Mater.* 55, 1547. (2020).
34. X. Xu, and N. Gupta, *J. Mater. Sci.* 54, 8401. (2019).
35. Xu. Xianbo, and N. Gupta, *JOM* 71, 4015. (2019).
36. P. Wang, E. Fan, and P. Wang, *Pattern Recogn. Lett.* 141, 61. (2021).
37. T. Sabiston, K. Inal, and P. Lee-Sullivan, *Compos. Sci. Technol.* 190, 108034. (2020).
38. J. Yang and H. Yao, *Extreme Mech. Lett.* 39, 100771 (2020).
39. K. Yanamandra, G.L. Chen, X. Xu, G. Mac and N. Gupta (2020), *Compos. Sci. Technol.* 198, 108318. (2020).
40. J. Kannala and E. Rahtu, *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)* (2012), pp. 1363–1366.

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.