Computational Materials Science 175 (2020) 109599

Contents lists available at ScienceDirect

Computational Materials Science

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

Predicting microstructure-dependent mechanical properties in additively manufactured metals with machine- and deep-learning methods

Carl Herriott[a], Ashley D. Spear[a,*]

[a] Department of Mechanical Engineering, University of Utah, Salt Lake City, UT, USA

ARTICLE INFO

Keywords:
Convolutional neural network
Machine learning
Metal additive manufacturing
Spatial-property prediction

ABSTRACT

In this work, we investigate the performance of data-driven modeling for mechanical property prediction of a simulated microstructural dataset. The dataset comprises realistic microstructural subvolumes of metal additive-manufactured stainless steel 316L and corresponding effective mechanical properties that were generated with a physics-driven modeling framework. The data-driven models leveraged for this work include Ridge regression, XGBoost, and a custom 3D convolutional neural network (CNN) based on VGGNet. Morphological and crystallographic features describing each microstructure serve as the inputs for the Ridge regression and XGBoost models. The CNN is trained with a 3D image of the microstructure represented by progressively informative input data (ranging from grain ID to crystal orientation, and supplemented with auxiliary features describing the mechanical loading) to determine the relative improvement among different feature types. Comparisons are drawn between the predictive performance of each data-driven model in terms of different scoring metrics and spatial-property maps. The computational efficiency of each data-driven model and the physics-driven modeling framework is also reported. Among all of the data-driven models tested, the CNN models that use crystal orientation as input (with or without auxiliary input features) provide the best predictions, require little pre-processing, and predict spatial-property maps in a matter of seconds.

1. Introduction

The development and use of physics-driven models to predict process-(micro)structure-property relationships in the context of metal additive manufacturing (MAM) has increased significantly in recent years. As two illustrative examples, Ahmadi et al. [1] and Andani et al. [2] generated MAM microstructures by repeatedly overlapping the geometry of a melt pool with a grain structure generated by Voronoi tessellation. While they did not use physics-driven modeling to predict the MAM microstructure, they did use physics-based mechanical simulations to investigate the mechanical properties of the simulated microstructures. In another example, Yan et al. [3] used a physics-driven approach to generate the solidified microstructure from a powder bed fusion process and to predict its mechanical properties via finite element modeling with a crystal-plasticity constitutive model. Despite the importance of physics-driven models for predicting relationships among process, microstructure, and mechanical properties for MAM, such frameworks generally suffer from excessive computational costs, which can be prohibitive for high-throughput analysis or simulation of large build domains. Recent process-microstructure-property frameworks (e.g., [4,5]) have found success in reducing computational expense by using a parallelized elasto-viscoplastic fast Fourier transform (EVPFFT) solver [6,7] to compute the full-field constitutive response over the microstructural volume of interest. Nonetheless, there remains a need to further accelerate predictions of microstructure and mechanical response for MAM to facilitate design engineering and certification/qualification procedures.

Data-driven modeling approaches, which include machine learning, provide a promising path toward expediting predictions of process-structure-property linkages. As stated in Ref. [8], "data-driven approaches aim to identify objectively (relying largely on the available data) the embedded correlations among selected inputs and outputs needed to study or model a given phenomenon." For clarity herein, we distinguish between machine- and deep-learning methods and models. *Machine learning (ML)* is a broad term and can encompass any supervised learning algorithm or model, such as basic linear regression, support vector machines, and random forests. *Deep learning (DL)* is considered a subset of ML and typically incorporates layers of artificial neuron-like connections–neural networks–to extract features and backpropagation of error to arrive at a prediction. The "deep" in deep learning refers to the number of successive layers within the models that extract higher-level features from the prior layers. DL models

[*] Corresponding author.
E-mail address: ashley.spear@utah.edu (A.D. Spear).

https://doi.org/10.1016/j.commatsci.2020.109599
Received 30 September 2019; Received in revised form 10 February 2020; Accepted 10 February 2020

usually extract their own features to arrive at a prediction; whereas, most ML models require the user to supply the relevant features for prediction. A convolutional neural network (CNN), a type of DL model, performs layers of convolutions with images and kernels to extract high-level features from the original image data. CNNs excel at extracting spatial information from an input. In terms of a metallic microstructure, this suggests that a CNN could capture grain-grain interactions and relationships, which might not otherwise be captured in user-defined input features supplied to more traditional ML models. Unless explicitly stated, "ML" will be used in the context of this paper to refer to any methods or models that do not use an artificial neural network for prediction. For more information regarding deep learning, the reader is referred to Refs. [9,10].

There are many recent examples in which data-driven models have been used for property prediction of 3D microstructures. For example, ML and DL methods have recently been applied to 3D microstructures for the prediction of mechanical properties using features from 2-point statistics as inputs [11,12], and using other information such as the crystallographic grain orientation as an input [13,14]. Cecen et al. [11] leveraged 2-point statistics with CNN-interpreted features to predict the properties of a relatively large dataset of synthetic, high-contrast composites. They were able to reduce their prediction error for the CNN by a factor of two when using 2-point statistics to augment the CNN-extracted features as opposed to just features selected by the CNN. Jung et al. [12] developed a database of synthetic microstructures using DREAM.3D and also applied 2-point statistics for microstructure quantification. They used a Gaussian process regression model and predicted the full-field response of their microstructures. Using their framework, they were able to determine an optimal microstructure for a given property, viz., a microstructure of dual-phase steel that will give the highest yield strength. In another example, Frankel et al. [13] developed a database of synthetic polycrystals, each with fewer than 30 grains. They built a hybrid DL model using a CNN encoder and a recursive neural network (RNN) to predict the elastic response and the onset of plasticity of their polycrystals. At that length scale, the microstructure plays a significant role in the variation in mechanical response. These few-grained polycrystals are somewhat analogous to the large-grained microstructures that can be created by MAM processes. Finally, Mangal [14] augmented a small dataset of synthetically generated microstructures and used 2D and 3D CNNs to predict hotspots, defined as grains with a von Mises stress above the 90th percentile. In that work, Mangal built several models based on popular CNNs, such as a fully convolutional network [15], Pixelnet [16], and Resnet [17]. Using various inputs, such as Euler angles and quaternions, Mangal achieved mixed results; the 3D CNNs were not able to optimize and make strong predictions, and the 2D models achieved moderate success. The mixed success is likely attributed to the limited size of the dataset and to models that were not fully tuned to the data. While the above examples provide motivation for applying ML and DL methods to predict properties of 3D microstructures, there remains a need to quantify the relative performance of different ML and DL models in terms of their abilities to predict effective mechanical properties in MAM microstructures.

In this research, we implement and compare the performances of different ML models (viz., Ridge regression and XGBoost) and DL models (viz., 3D CNNs) based on predictions of location-dependent effective mechanical properties throughout simulated MAM microstructural domains. The ML and DL models are applied to an expansive dataset of realistic MAM microstructures (approximately 7700) generated in previous work by the authors using a high-fidelity, physics-based modeling framework [4]. Volume-averaged microstructural descriptors are used as inputs to the ML models; whereas, 3D image data in the form of basic microstructural information (e.g., grain IDs) are input to the CNN model. A secondary objective of this work is to determine the relative improvement of the CNN-model predictions when the training data include progressively informative features, including crystal orientations, loading direction, and micromechanical Taylor factor [18]. Because 3D CNNs account inherently for the spatial relationships embedded within the data, we test the hypothesis that the CNN models will have improved predictions over the more traditional ML models because of their ability to learn the relationships between 3D microstructural arrangements and corresponding mechanical properties.

2. Methods

In this section, we first describe the high-fidelity, physics-based simulations used to generate the training data for the ML and DL models. Effective mechanical properties computed from physics-based simulations of additively manufactured SS316L are treated as the target output. Sections 2.2 and 2.3 describe the ML and DL models, respectively, including the structure and types of inputs that are used for each model. Finally, the method for evaluating the ML and DL model predictions is presented in Section 2.4.

2.1. Physics-driven modeling framework used to derive training data

The multi-scale, multi-physics modeling framework used to develop the dataset for this work was introduced in previous work by the authors [4]. For completeness, and to provide context for the training data referenced in Section 2.2 and onward, a brief overview of the framework and its application to additively manufactured SS316L is presented in this section. The framework leverages both MAM process modeling along with microstructure-sensitive, solid-mechanics modeling to generate realistic microstructures and calculate their effective mechanical properties, respectively. Any unspecified parameters, such as the thermal properties and cubic elastic constants for SS316L, can be found in Ref. [4].

2.1.1. Process model

Thermal model: In previous work, a macroscale finite-volume model was used to predict the thermal history of a direct laser deposition (DLD) process. The metal-gas interface was captured by a level-set function, as outlined in [19], and fluid flow in the molten pool was ignored. The laser power (represented by a Gaussian distribution) was set to 200 W, with a $1/e^2$ beam diameter of 440 μm and a scanning velocity of 26 mm/s. The hatch spacing was 250 μm. The three axes of the computational build domain are labeled as the scanning direction (SD), build direction (BD), and the transverse direction (TD). The simulation domain for the thermal model used an element size of 25 μm. The simulated laser scanned in an alternating pattern along the SD direction, with the starting direction of the laser rotating by 180° every successive layer.

Grain-growth model: The resultant thermal history was utilized by the mesoscale grain-nucleation and grain-growth simulation, which employed a 3D cellular automata (CA) model [20]. Both epitaxial and bulk nucleation were considered. Epitaxial nucleation occurs when the temperature of a cell along the fusion line drops below the liquidus temperature. A nucleus forms in this cell with the same orientation as a neighboring cell in the previous layer. Bulk nucleation is a function of the maximum nuclei density, N_0; the critical undercooling value, ΔT_c; and the standard deviation of the undercooling, ΔT_σ. Varying the nucleation parameters affects the amount of grain nucleation and the resulting grain structure in the build, as detailed in [19]. The grain growth post nucleation was determined by the CA method, which assumed the growth velocity as a function of the local undercooling. A polynomial fit developed from [21] was used to approximate the growth velocity.

Four total builds were simulated using the method outlined above. Each build was simulated with varying grain-nucleation parameters. While the standard deviation of the undercooling, ΔT_σ, was found to have little effect [19], N_0 and ΔT_c were varied such that the microstructures range from fully columnar to fully equiaxed. The four build

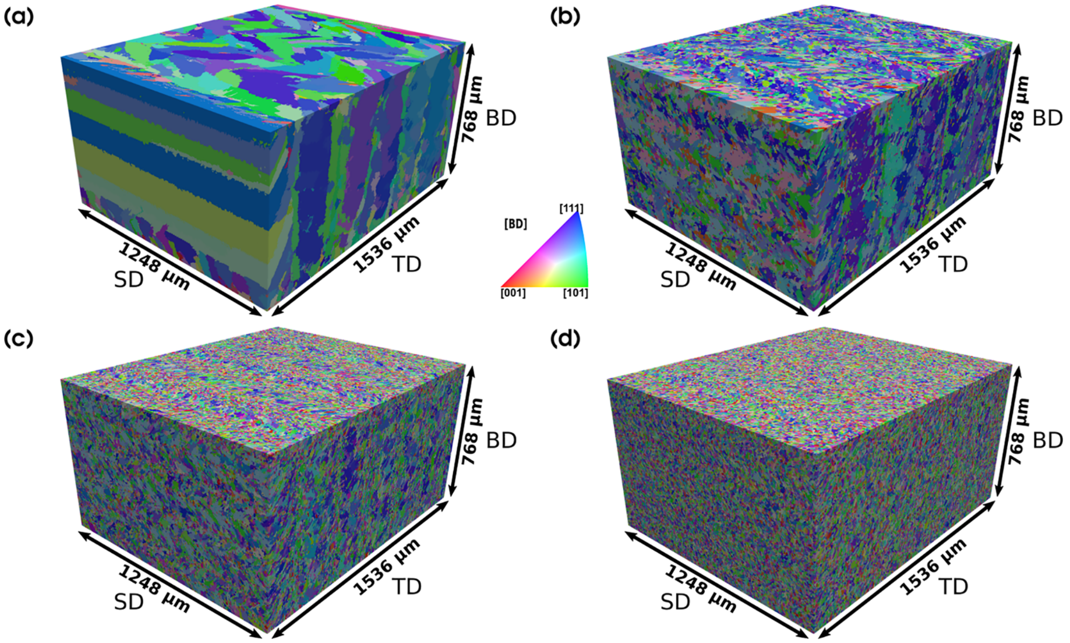

Fig. 1. A 3D representation of each simulated microstructure generated by varying the nucleation parameters in the multi-scale, multi-physics modeling framework reported previously by the authors [4]. Figures (a)–(d) refer to Domains A–D, respectively. Data from the four domains are used as training and testing data in the current work.

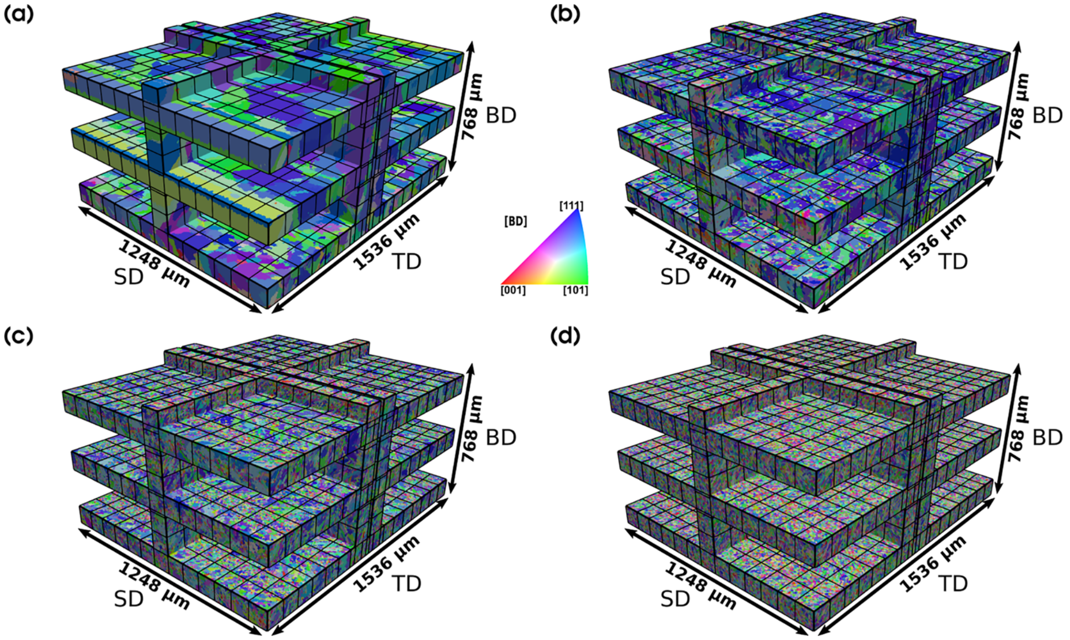

Fig. 2. A 3D representation of the sampling layers and subvolumes defined for each domain, as reported previously by the authors [4]. Figures (a)–(d) refer to Domains A–D, respectively. The individual subvolumes serve as the data used for training and testing the data-driven models in this work. For Domains A and D, the middle sampling layer in the TD-SD plane is treated as a holdout layer used to test the performance of the data-driven models in predicting an entire property map for the respective sampling layer.

domains, referred to as domains A-D, are shown in Fig. 1. The predictive capability of this grain-nucleation and grain-growth model has been qualitatively validated against experimental MAM microstructures in Ref. [19].

2.1.2. Solid-mechanics model

Subvolume sampling: Once the four build domains were generated, each one was divided into six sampling layers of interest (see Fig. 2), which were then further discretized into individual subvolumes that served as the input to the crystal-plasticity modeling portion of the framework. Each subvolume is $96 \times 96 \times 96\ \mu\text{m}^3$ (approximately the same height as a build layer) with a voxel size of 3 μm. Each voxellized subvolume was passed through DREAM.3D, which was used to quantify microstructural features (detailed in 2.2.1) and to write the input for the EVPFFT model. Note that the three layers in the TD-SD plane shown in Fig. 2 were chosen to investigate the variability within an AM domain with respect to the build direction. The two sampling layers in the SD-BD plane, one centered along a laser-scan track and the other

Table 1
List of all microstructural descriptors used as inputs to the Ridge regression and XGBoost models. Except for the texture index (T), each descriptor is converted to a grain-volume-weighted average and standard deviation, resulting in two scalar values per descriptor, or 35 descriptors in total per microstructural subvolume.

Feature Name	Symbol	Description
Volume	V	Average grain volume
Axis lengths	$a,\ b,\ c$	Semi-axis lengths of a best-fit ellipsoid for each grain
Aspect ratio	$b/a,\ c/a$	Ratios of the semi-axis lengths
Number of neighbors	N_N	Number of neighboring grains for a given grain
Equivalent spherical diameter	ESD	Diameter of a sphere with a volume equivalent to that of a given grain
Omega-3	Ω_3	3rd invariant of the second-order moment-of-inertia tensor
Surface area to volume ratio	$SA{:}\ V$	Ratio of grain surface area to grain volume
Average distance to grain boundary	d_{GB}	Distance to the nearest grain boundary for a given grain
Average distance to triple junction	d_{TJ}	Distance to the nearest triple junction for a given grain
Average distance to quadruple point	d_{QP}	Distance to the nearest quadruple point for a given grain
Schmid factor	m	Schmid factor of the slip system with the highest value for each grain
Micromechanical Taylor factor	M^{micro}	Accumulated local crystallographic shear normalized by the local von Mises strain, averaged for each grain
Average misorientation	ω_{Avg}	Average misorientation of neighboring grains to a given grain
Texture index/strength	T	L-2 norm of the orientation distribution function (a measure of how strongly a subvolume is textured)
Average slip transmission factor (m-prime)	m'	A factor that measures the ease of slip transmission across a given grain boundary

between two successive tracks, were selected to explore any variability that might appear between and along scan lines.

MASSIF/EVPFFT: The elasto-viscoplastic fast Fourier transform code, also known as Micromechanical Analysis of Stress-Strain Inhomogeneities with fast Fourier transforms (MASSIF), is a computationally efficient, full-field crystal-plasticity solver, originally developed by Lebensohn et al. [6]. MASSIF uses the Voce hardening law to update the critical resolved shear stress (CRSS) at each time step and for all slip systems at every point in the model. The MASSIF code is parallelized and uses the HDF5 file format for faster I/O speeds. A modification was made to the MASSIF code to account for grain-boundary strengthening, which was initially reported, along with the MASSIF formulation, in Ref. [4]. All subvolumes were loaded uniaxially to 1% strain in each of the two in-plane directions for a given sampling layer of interest.

2.1.3. Site-specific property maps

The full-field response of each microstructural subvolume was homogenized to extract an effective stress-strain curve. A modified Hough transform code was applied to the stress-strain curve to calculate the effective Young's modulus and the effective yield strength, for each direction of the applied displacement. These effective mechanical properties were then visualized in the form of a heat map, or contour plot. The property maps were qualitatively validated in previous work [4] by comparison with experimental observations from the literature. While ongoing work focuses on quantitative validation, the qualitatively validated property maps from the physics-driven modeling framework are treated as the reference solution to which all ML- and DL-model predictions are compared.

2.1.4. Raw (unprocessed) dataset passed to the machine- and deep-learning models

The resulting dataset that serves as the basis for training the ML and DL models in the current work consists of the voxel-based microstructure for each subvolume, along with the corresponding effective mechanical properties (viz., effective yield strength) for specific loading directions, as predicted by the high-fidelity, multi-physics modeling framework presented above. For the ML and DL models, the raw data (i.e., the $96 \times 96 \times 96\ \mu m^3$ voxellized subvolumes) are then subjected to feature extraction, which is described in the next section. Each of the four MAM build domains comprises 960 unique microstructural subvolumes, each with a corresponding effective yield strength in the two in-plane directions of the sampling layer (e.g., a property map in the TD-SD plane includes effective yield strength values in the TD and SD directions). This gives a total of 1920 subvolume/mechanical property combinations per microstructural domain, resulting in 7680 data points total among the four MAM build domains. To the authors' knowledge, this is currently one of the largest datasets of its kind used for mechanical-property prediction of 3D microstructures with machine and deep learning. Other datasets have ranged from 1100 to 5900 microstructures [11,12,14]. It is also the only dataset for which the input microstructures were simulated using a physics-based model, and not synthetically generated by a program such as DREAM.3D [22].

2.2. Machine learning methods: Ridge regression and XGBoost

2.2.1. Feature extraction and data pre-processing

Various morphological and crystallographic descriptors are extracted to parameterize the microstructure in each subvolume and to serve as inputs to the Ridge regression and XGBoost models. The full list of features is found in Table 1, and select features are detailed here. The Schmid factor relates the tensile axis to the crystallographic slip plane normal and is defined as:

$$m = cos(\lambda)cos(\phi), \tag{1}$$

where λ is the angle between the global tensile axis and a given slip direction within the slip plane, and ϕ is the angle between the tensile axis and the slip plane normal. The Schmid factor is typically used to determine which slip plane and direction will resolve the highest shear stress (i.e., which slip system will first meet the condition for slipping, assuming a fixed CRSS and global loading direction). Since SS316L is face-centered cubic (FCC), 12 Schmid factors are calculated for each grain, one for each slip system. The Schmid factor with the highest value is then assigned to each voxel within that grain. The slip transmission factor measures the compatibility between active slip systems in neighboring grains [23]. The factor is represented as the dot product of the angle between slip directions in neighboring grains with the angle between the slip plane normals. The micromechanical Taylor factor is calculated during simulated deformation and is quantified as the accumulated local crystallographic shear normalized by the local von Mises strain [18]. The average distance features are calculated by measuring the Euclidean distance (in voxels) to the nearest grain boundary, triple junction, and quadruple point for each voxel. Then, for a given grain, the values of all cells (voxels) within that grain are averaged.

All features used by the Ridge regression and XGBoost models, except the texture index and the micromechanical Taylor factor, are extracted as per-voxel or per-grain values using DREAM.3D's suite of crystallographic analysis filters [22]. MTEX [24] is used to calculate the orientation distribution function (ODF) and corresponding texture index of each subvolume. The voxel-based values of micromechanical Taylor factor computed using MASSIF (taken at the final loading step) are converted to grain-averaged values. Finally, grain volume-weighted averages[1] are calculated to obtain a single scalar value for each feature to describe a given microstructural subvolume. For every feature except the texture index, volume-weighted Gaussian standard deviations are also calculated and used as additional features to capture the distribution within a given subvolume. Before being passed into an ML model, each feature is standardized by subtracting the feature's mean and dividing by the standard deviation. In total, 35 different features are used to parameterize each subvolume and serve as the inputs to the Ridge regression and XGBoost ML models. We note that the micromechanical Taylor-factor feature is based on the MASSIF simulation of each subvolume, so its use is impractical in the context of replacing high-fidelity numerical simulation with ML models. It is included to provide a qualitative verification of the ML-model predictions; that is, we expect to see strong improvement in predictions of effective yield strength after adding the micromechanical Taylor factor to the training data due to the fundamental relationship between them. The results with and without including the micromechanical Taylor factor are presented in Section 3.1.

2.2.2. Machine-learning regression models

Ridge regression is implemented via the Scikit-Learn Python distribution [25]. Ridge, or Tikhonov, regression uses linear least squares with L2 regularization [26]. Ridge regression seeks to minimize the following objective function:

$$\mathscr{L} = ||Xw - y||_2^2 + \alpha||w||_2^2, \tag{2}$$

where X are the inputs, y are the targets, w are the coefficients or weights, and α is a tunable regularization parameter. Here, X are the features defined in Table 1 and y are the target values of effective yield strength for the subvolumes used in training. Ridge regression, which is typically used when the data suffer from multicollinearity, is chosen here to serve as a basic model comparison. Fivefold cross validation is used to fit the regularization penalty parameter, α. A final (tuned) value of 0.35 is selected for α.

XGBoost is a popular gradient-boosting ML package [27]. Its popularity comes from its speed [27], effectiveness [28], and compatibility with Python and Scikit-Learn. XGBoost minimizes its objective function by ensembling multiple decision trees and adding the tree predictors to correct prior models. The weights are updated after each iteration via gradient descent. During hyperparameter tuning, a fivefold cross-validation randomized search method is used to help reduce overfitting [29]. For each cross-validation fold, the search function generates a specified number (in this case, 100) of random sets of hyperparameter combinations and evaluates the model using those random sets. The hyperparameters to be tuned include maximum depth, number of trees, minimum samples per leaf, and the minimum samples per split. The final (tuned) values for the aforementioned hyperparameters are found to be 6, 100, 3, and 2, respectively.

The Ridge regression and XGBoost models serve as benchmark comparisons to test whether the CNN models perform better due to their ability to extract spatial information from raw image data.

2.3. Deep learning method: convolutional neural network

2.3.1. CNN input features

Unlike the ML models described above, the CNN model does not require extraction of subvolume-based features to parameterize a given microstructure. Rather, the input is based on a 3D image of the microstructure. In this study, the 3D images passed to the CNN are simply the voxellized, microstructural subvolumes along with their corresponding effective mechanical properties derived from the physics-driven modeling framework. One of the objectives of this study is to assess how the performance of the DL model changes when trained with progressively informative input data. The most basic description requires only the image data of the microstructural subvolume (either grain ID or crystal orientation for each voxel); a more informative description also includes information about boundary conditions (e.g., loading direction); and the most informative description includes metrics computed from the crystal-plasticity simulations (e.g., micromechanical Taylor factor). To test the relative performance, eight different CNN models (all having the same architecture) are trained with different voxel-based feature data, or channels. Four CNN models are trained with grain ID as the primary channel, and four are trained with crystal orientation, represented by quaternions, as the primary channels. The number of input channels can vary depending on the combination of the primary feature data with or without auxiliary features, described next.

Of the four models having either grain IDs or quaternions as the primary channel(s), one model has an additional "one-hot" vector describing the global loading direction applied to the microstructural subvolume. Since the loading vector is constant for an entire subvolume, it is incorporated immediately before the fully connected layer in the CNN architecture (described in the next subsection), as opposed to having its own channel during convolution.

Another of the four CNN models has, in addition to grain IDs or quaternions, the Schmid factor as an input channel. The Schmid factor for each voxel is calculated according to the method described in Section 2.2.1.

Finally, one of the four CNN models has, in addition to grain IDs or quaternions, the micromechanical Taylor factor (taken at the last loading step) as an input channel. As mentioned previously for the ML models, it is counterproductive from a prediction/application standpoint to train the DL model using features derived from the crystal-plasticity simulation, given that the overall objective of using the ML/DL models is to circumvent the expensive physics-based simulations to make rapid forward predictions. However, the purpose of doing so here is twofold: 1) to verify the CNN model predictions, because we know that the micromechanical Taylor factor is directly related to the effective yield strength of the microstructure, and 2) to quantify relative performance of the previously mentioned CNN models, which are trained using data that do not require crystal-plasticity calculations. All CNN data are standardized before use in training.

2.3.2. Convolutional neural network architecture

The CNN architecture built for this work is a 3D implementation of the first three convolutional blocks of VGGnet [30], with four fully connected (FC) layers at the end. Each convolutional layer uses a $3 \times 3 \times 3$ kernel size and He normal weight initialization [31]. A rectified linear unit (ReLU) activation [32] is used for all convolution and FC layers. After the convolutional section, a dropout layer [33] with a rate of 0.75 is used. The first two FC layers contain 1024 hidden units, followed by 512 hidden units before the final output layer. No layers were pretrained. Because this is a regression task, the model seeks to minimize the mean-squared logarithmic error. The Adam [34] optimization function with AMSGrad [35] enabled is used with an initial learning rate of 0.0005. The learning rate is reduced by half upon the plateauing of the validation loss. The CNN stops training if the validation loss does not improve after 11 epochs.

The input subvolumes are of size $32 \times 32 \times 32$ voxels, or $96 \times 96 \times 96\ \mu\text{m}^3$, and contain one to five channels depending on what combination of primary and auxiliary features are used, as described in the previous subsection. As previously mentioned, since the one-hot loading direction feature does not change spatially, it is added to the model after the final convolution and before the FC layers. Fig. 3 depicts the CNN architecture developed for this work.

[1] The volume of a grain is calculated by multiplying the volume of a voxel, $3 \times 3 \times 3\ \mu\text{m}^3$, by the number of voxels associated with that grain.

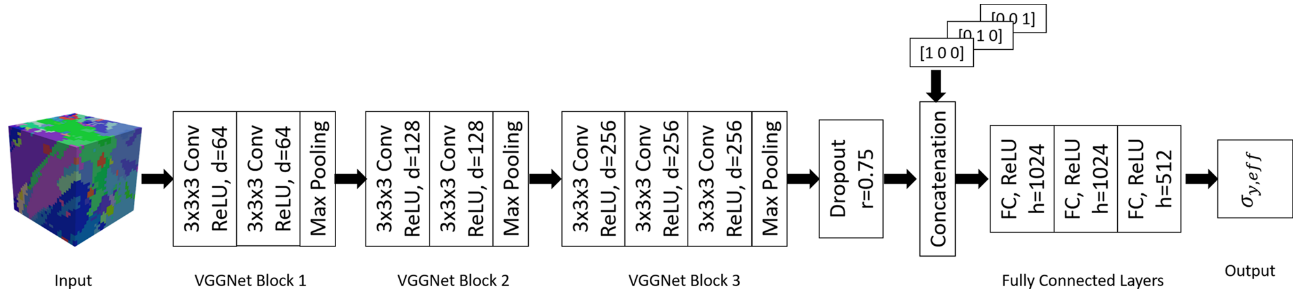

Fig. 3. Architecture of the 3D convolutional neural network based on VGGNet. The concatenation layer is optional and is only used when the load direction vector is included.

2.4. Evaluation of ML- and DL-model predictions

To evaluate the performance of the ML and DL models, two scoring metrics, the root mean squared error (RMSE) and R^2 value, are calculated based on the reference values of effective yield strength computed using MASSIF (see Section 2.1). Scores for the ML and DL models are calculated separately for a holdout dataset (832 data points) and for the remaining dataset (6848 data points), the latter of which is divided into a randomized 80/20 train/test split. It is noted that during the training phase for the CNN, 10% of the training data is set aside for model validation to help prevent overfitting. The train/test split data are also stratified such that each split contains a proportionate number of subvolumes from each build domain. The holdout dataset, which comprises all subvolumes in the middle sampling layer of the TD-SD plane for both Domain A and Domain D (shown in Fig. 2), is used to investigate through blind prediction how well the ML and DL models predict the spatial variability of the effective mechanical properties. These two MAM build domains are chosen for the blind predictions because they represent the microstructural extremes of the AM SS316L dataset–fully columnar and fully equiaxed grain structures, respectively. The inputs to the Ridge regression and XGBoost models are the feature vectors described in Section 2.2.1, and the CNN inputs are the 3D microstructure images detailed in Section 2.3.1. Once trained, each ML and DL model is used to predict a spatial-property map for the two holdout layers, where the property shown is the effective yield strength in the transverse direction. The property maps for the holdout layers in Domains A and D are then directly compared to the corresponding maps generated using crystal-plasticity modeling.

3. Results and discussion

3.1. Ridge regression and XGBoost predictions

Table 2 provides the scores for the Ridge regression and XGBoost models trained using the features listed in Table 1, either excluding or including the micromechanical Taylor-factor feature. Focusing on the performance metrics for just the holdout set (i.e., the blind prediction for one entire sampling layer in Domains A and D) and excluding the micromechanical Taylor-factor feature, the Ridge regression model performs marginally better than the more complex XGBoost model. On the 80/20 test-split data, the XGBoost model performs marginally better than the Ridge regression model. However, neither model seems to outperform the other significantly in terms of R^2 or RMSE values. As expected, the addition of the micromechanical Taylor-factor feature makes a marked improvement to both models.

Table 2
Scores for the Ridge regression and XGBoost models when predicting the effective yield strength for the MAM microstructure dataset.

Machine-Learning Models		Dataset			
		80/20 Split		Holdout set	
	Input features	R^2	RMSE (MPa)	R^2	RMSE (MPa)
Ridge Regression	Without M^{micro}	0.74	14.66	0.77	20.03
	With M^{micro}	0.86	10.64	0.93	10.85
XGBoost	Without M^{micro}	0.79	13.14	0.74	21.12
	With M^{micro}	0.90	9.172	0.94	10.01

Figs. 4 and 5 provide visual comparisons of the property maps predicted by the different ML models for the holdout layer in Build Domains A and D, respectively. In each figure, the corresponding property map based on the high-fidelity, crystal-plasticity simulations (see Section 2.1) is also shown for reference. The contour limits corresponding to the maps of effective yield strength are set to those of the reference map. For Domains A and D (Figs. 4 and 5, respectively), the two types of ML models show similar predictions in the spatial distributions of effective yield strength, which is consistent with the similarities in performance metrics reported in Table 2. Both ML models are also able to recover the banded pattern in Domain D. However, neither model does particularly well at predicting the extremes of the MASSIF-simulated yield strength range. As anticipated, the addition of the micromechanical Taylor factor recovers the trends of the MASSIF-simulated maps, with the XGBoost model capturing more of the hot spots of lower yield strengths than the Ridge regression model.

3.2. Convolutional neural network predictions

The CNN-model scores for the predictions of effective yield strength are provided in Table 3. Scores for eight different CNN models are presented, where each model was trained using either grain ID or crystal orientation as the primary channel(s), and, in six cases, an auxiliary channel (Schmid or micromechanical Taylor factor) or a one-hot vector representing the global loading direction. These results are discussed further in Sections 3.3 and 3.4.

Figs. 6 and 7 provide visual comparisons of the property maps predicted by the different DL models for the holdout layer in Domains A and D, respectively. Figs. 6a and 7a show the property maps from the high-fidelity, crystal-plasticity simulations for reference. Figs. 6b–e and 7b–e show the CNN-predicted maps for the two build domains when grain ID is used as the primary feature, and Figs. 6f-i and 7f-i show the CNN-predicted maps when quaternions are used as the primary feature. In all cases, the contour limits corresponding to the maps of effective yield strength are set to those of the reference map. The differences among the CNN-predicted maps are discussed further in the next two subsections.

3.3. Comparison between machine- and deep-learning predictions of effective mechanical properties

The results presented above are now synthesized to compare the ML and DL predictions of effective yield strength. When comparing the ML and DL models, focusing on predictions for the holdout dataset and

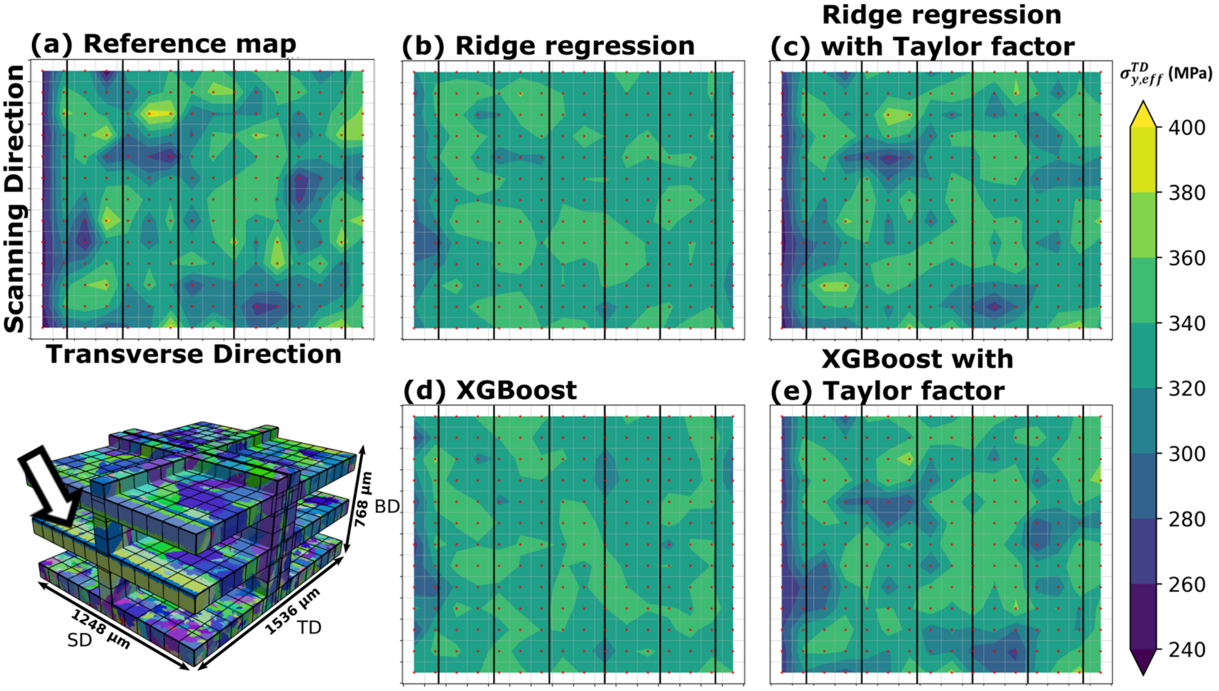

Fig. 4. Comparison of property maps (viz., effective yield strength in the transverse direction) predicted by different machine-learning models for the holdout layer indicated in Build Domain A: (a) for reference, property map generated using crystal-plasticity modeling [4]; (b) Ridge regression, excluding M^{micro} from training set; (c) Ridge regression, including M^{micro} in training set; (d) XGBoost, excluding M^{micro} from training set; (e) XGBoost, including M^{micro} in training set. Vertical black lines correspond to laser scan tracks.

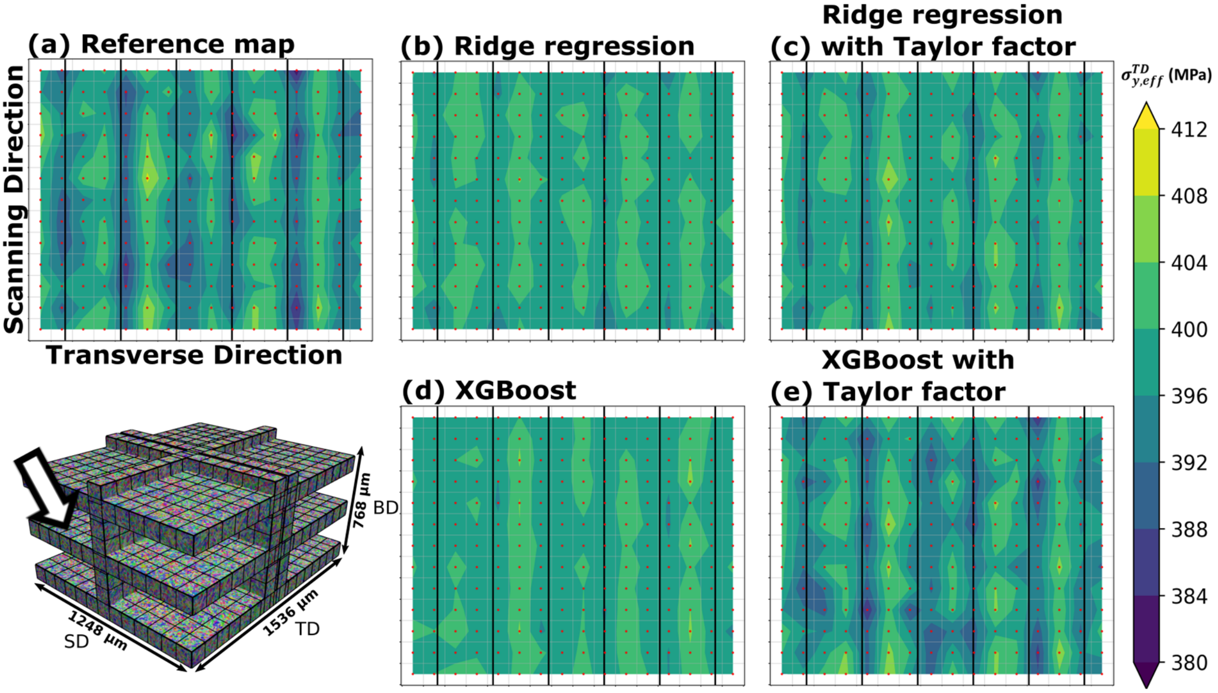

Fig. 5. Comparison of property maps (viz., effective yield strength in the transverse direction) predicted by different machine-learning models for the holdout layer indicated in Build Domain D: (a) for reference, property map generated using crystal-plasticity modeling [4]; (b) Ridge regression, excluding M^{micro} from training set; (c) Ridge regression, including M^{micro} in training set; (d) XGBoost, excluding M^{micro} from training set; (e) XGBoost, including M^{micro} in training set. Both models are able to recover the banded pattern present in (a). Vertical black lines correspond to laser scan tracks.

excluding the micromechanical Taylor-factor feature, the scores for the CNN model trained using grain IDs (either alone or with an additional feature) are lower than those for either of the two ML models (R^2 range of 0.68–0.70 compared to a range of 0.74–0.77). However, when the CNN model is trained using quaternions as the primary channels (either alone or with an additional feature), the scores are higher than those for either of the two ML models (R^2 range of 0.81–0.86 compared to a range of 0.74–0.77). Interestingly, when the CNN model is trained using only quaternions, without any other information, the CNN model achieves an R^2 value of 0.84 and an RMSE value of 16.57 MPa based on blind predictions of the holdout layers in two different MAM build domains.

Table 3
Scores for the CNN models when predicting the effective yield strength for the MAM microstructure dataset.

Deep-Learning Models			Dataset			
			80/20 Split		Holdout set	
	Primary Feature	Auxiliary Feature	R^2	RMSE (MPa)	R^2	RMSE (MPa)
CNN	Grain ID	–	0.70	15.81	0.68	23.45
		Load vector	0.76	14.10	0.70	22.70
		Schmid factor	0.69	15.89	0.68	23.67
		M^{micro}	0.87	10.32	0.94	10.08
	Quaternions	–	0.80	12.78	0.84	16.57
		Load vector	0.83	11.85	0.86	15.53
		Schmid factor	0.79	13.26	0.81	18.15
		M^{micro}	0.90	9.206	0.95	9.230

For Build Domain A (Figs. 4 and 6), ignoring the models that include the micromechanical Taylor factor, the CNN models that are trained using quaternions predict maps of effective yield strength that are much more similar to the reference map than any of the other ML or DL predictions. Unlike the ML-model predictions (Fig. 4), the CNNs trained with quaternions are able to predict the general patterns of effective yield strength, even without including any auxiliary features. While some of the CNN models are able to capture the general patterns in the property map of Domain A, neither the ML nor DL models, in the absence of the micromechanical Taylor factor, are able to capture the extreme values of yield strength for that domain. It is anticipated that more training data would further improve the predictions.

For Build Domain D (Figs. 5 and 7), ignoring the models that include the micromechanical Taylor factor, both the ML models and the DL models trained using quaternions are able to predict the banded pattern that is apparent in the reference property map. However, the CNN models are better able to match the range of yield strength values present in the holdout layer for Build Domain D than are either of the ML models. It is worth noting that the periodicity with respect to the laser scan lines, resulting in the banded pattern shown, has been observed experimentally under certain MAM build conditions. For example, Hayes et al. [36] observed strongly textured and qualitatively similar bands that formed in a Ti-6Al-4V volume produced via wire-fed directed energy deposition. Further comparison between experimental observations and the simulated property maps that serve as the reference solution (i.e., from physics-driven modeling) can be found in Ref. [4].

To summarize the comparison between the ML- and DL-model predictions, the ML models (Ridge regression and XGBoost) outperform the DL models (i.e., CNNs) when the latter rely on grain IDs as the primary input; however, the DL models that use crystal orientation as the primary input significantly outperform all other models considered, achieving an R^2 value upwards of 0.86 in blind predictions of the holdout layers for two different MAM build domains. Furthermore, the DL models offer the benefit of requiring very little pre-processing and feature extraction, instead allowing the model, itself, to extract relevant higher-level features from the image data.

3.4. Effect of input-data type on CNN predictions

We now focus on the results from Table 3 and Figs. 6 and 7 to investigate the effect of using progressively informative features on the prediction performance of the DL models. First, we focus on CNN predictions made using grain IDs as the primary input channel. One interesting observation is that the degree of spatial variability of properties predicted by the CNN models trained using grain IDs as the primary feature is vastly different between Domains A and D. Notice that the CNN predictions in Fig. 6b-d for Domain A exhibit very little spatial variability; whereas, the predictions by the same CNN models for Domain D (Fig. 7b–d) exhibit significant variability, with some predictions falling outside the range of the reference yield-strength values. The ability of a given CNN model to discern spatial variability in properties is different between Domains A and D likely because of the vastly different grain structures between the two domains and because there is a greater variance of input values (grain IDs) in Domain D than in Domain A.

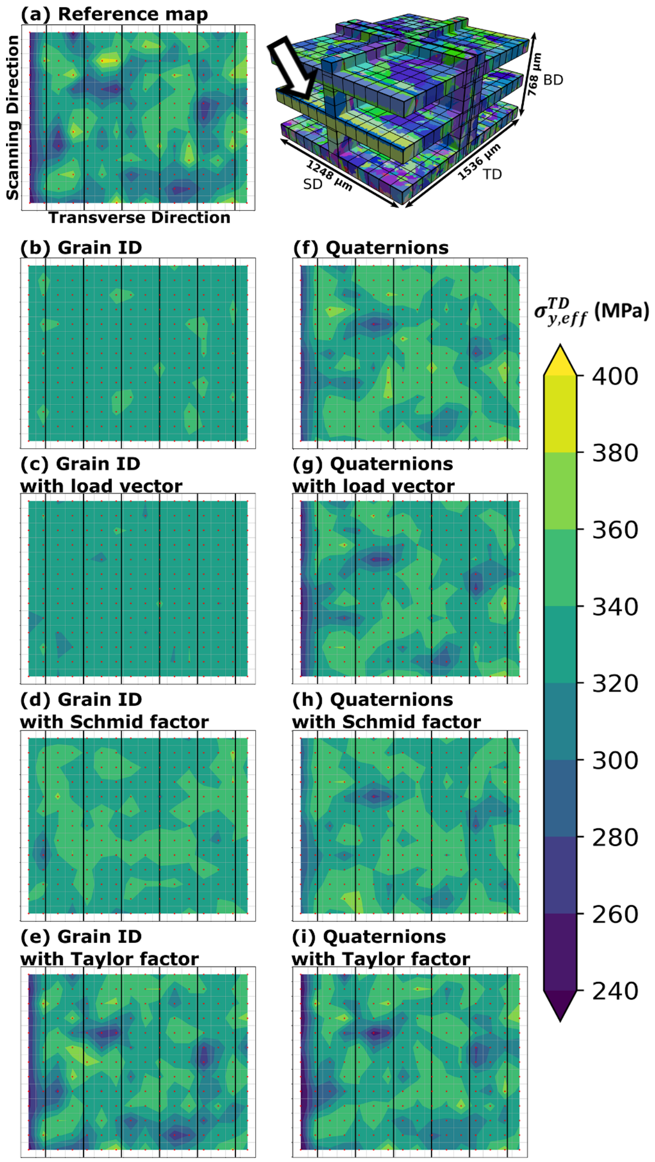

Fig. 6. Comparison of property maps (viz., effective yield strength in the transverse direction) predicted by deep-learning models for the holdout layer indicated in Build Domain A: (a) for reference, property map generated using crystal-plasticity modeling [4]; (b–e) CNNs with grain ID as the main feature in the training data; (f-i) CNNs with quaternion as the main feature in the training data. Auxiliary features used to train each model include: (b,f) none; (c,g) one-hot vector representing the global loading direction; (d,h) Schmid factor; (e,i) micromechanical Taylor factor. Vertical black lines correspond to laser scan tracks.

On the other hand, the spatial variability of mechanical properties is much more consistent between Domains A and D for the CNN models trained using quaternions as the primary input. In fact, the performance of the CNN models trained using quaternions is significantly better than that of any corresponding CNN model trained using grain IDs. This is evident from both the property maps shown in Figs. 6 and 7 and from

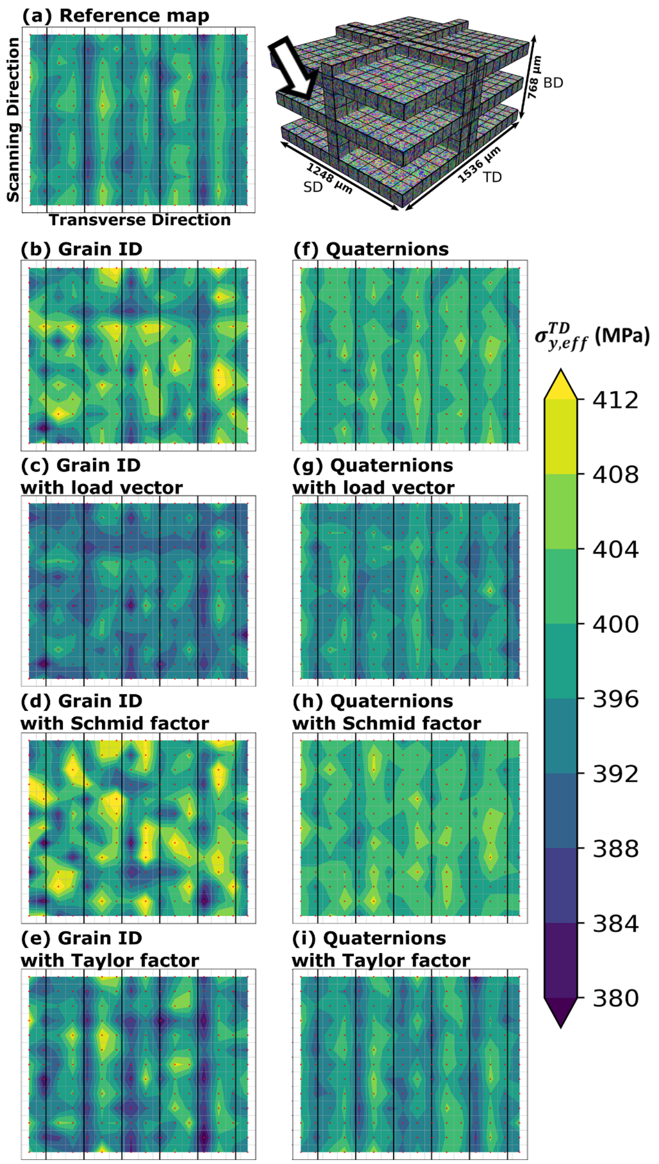

Fig. 7. Comparison of property maps (viz., effective yield strength in the transverse direction) predicted by deep-learning models for the holdout layer indicated in Build Domain D: (a) for reference, property map generated using crystal-plasticity modeling [4]; (b–e) CNNs with grain ID as the main feature in the training data; (f-i) CNNs with quaternion as the main feature in the training data. Auxiliary features used to train each model include: (b,f) none; (c,g) one-hot vector representing the global loading direction; (d,h) Schmid factor; (e,i) micromechanical Taylor factor. Vertical black lines correspond to laser scan tracks.

the scores reported in Table 3. In general, the improved performance and more consistent predictions of spatial variability between build domains for CNN models trained with quaternions confirms that, unlike grain IDs, crystal orientations provide a physically based input from which relevant information about crystallographic arrangements and textures can be extracted and linked to mechanical properties.

The most fundamental representation of the microstructure is the image depicting the spatial arrangements of grains, and we have just shown that crystal orientations lead to better predictions of mechanical properties than simply using grain IDs to represent the spatial arrangement of grains, but what is the impact of including more informative features pertaining to the mechanical loading conditions? The most basic representation of the applied boundary conditions is a vector describing the axis of uniaxial loading, which is treated as a one-hot vector input just before the multi-layer perceptron (see Fig. 3). As shown in Table 3, whether the primary input is an image of grain IDs or crystal orientations, there is a minor improvement in predictions of effective yield strength if the loading vector is also included. Although minor (R^2 improvement of 0.02 in the case of the holdout-layer predictions), the improvement in model performance might be worth the negligible cost of including the vector. A more costly representation of the loading is the Schmid factor, which is calculated per grain and input as a separate channel prior to convolution. Interestingly, including the Schmid factor leads to either no difference or to a slight *decrease* in the CNN-prediction scores. It appears that incorporating the Schmid factor somehow obfuscates the relevant relationships between the crystallographic arrangements and the effective yield strength. One possible explanation for this is that max-pooling during convolution (see Fig. 3), combined with the fact that the Schmid factor does not account for grain-grain interactions, leads to irrelevant or unrepresentative information being passed to the multilayer perceptron (i.e., the fully connected layers). However, this idea remains to be tested. The most informative and computationally expensive metric that introduces the mechanical loading into the DL model is the micromechanical Taylor factor, which is computed from the high-fidelity, crystal-plasticity simulations described in Section 2.1. Including the micromechanical Taylor factor as an input channel, in addition to either grain IDs or quaternions, leads to R^2 values for the holdout dataset of 0.94 and 0.95, respectively. While it is considered impractical to include results from the crystal-plasticity simulations as input to the DL model, the result provides verification that the CNN model is mapping the inputs to the target outputs in an expected manner.

3.5. *Implications of ML and DL models in microstructure-property predictions*

The primary motivation for exploring the use of ML and DL models in this work is to expedite the prediction of microstructure-sensitive mechanical properties for MAM compared to more expensive simulations involving physics-based constitutive models. For comparison, the time required to simulate one microstructural subvolume (up to 1% strain) is approximately 185 s on eight CPUs using the MASSIF framework described in Section 2.1. Given that there are 208 individual subvolumes comprising a single layer in the TD-SD plane of a given build domain, the time required to generate an entire map of effective yield strength for one of the holdout layers reported above is approximately 85 CPU hours using MASSIF. Training the ML models requires approximately three minutes for XGBoost and three minutes for Ridge regression (although, the majority of this time is spent on loading the data into memory). For the CNN models, training requires approximately 99 s per epoch on an NVIDIA GeForce GTX 1070 machine. While runtime for training the CNN can vary because early stopping is used, most models are trained in one to four hours. Remarkably, once trained, each of the ML and DL models predicts the complete map of effective yield strength for a given holdout layer in less than two seconds on a single processor. Such rapid predictions have significant implications for high-throughput explorations of MAM design space and possibly for real-time prediction of properties during the MAM process.

While this work focuses on applying ML and DL models to improve the efficiency of the structure-to-property predictions for MAM microstructures, there is a significant opportunity to apply similar approaches to improve the efficiency of process-to-structure predictions for MAM. The ultimate aim is to enable rapid and reasonably accurate process-structure-property predictions to support MAM design, optimization, qualification, and certification.

4. Conclusions

This work investigates the ability of machine-learning (ML) and deep-learning (DL) models to predict microstructure-sensitive mechanical properties in metal additive manufacturing (MAM) using results from high-fidelity, multi-physics simulations as training data. In previous work [4], four AM build domains exhibiting vastly different microstructures of SS316L were simulated using crystal-plasticity modeling to generate maps of effective mechanical properties (viz., effective yield strength in a given loading direction). The resulting microstructural subvolumes and corresponding yield-strength values (approximately 7700 data points in total) are used in this work to train two types of ML models (Ridge regression and XGBoost) and one type of DL model (CNN). The ML models require substantial pre-processing to extract volume-averaged descriptors that describe the microstructure of a given subvolume; whereas, 3D image data in the form of basic microstructural information (e.g., grain IDs) are input to the CNN model, from which higher-level features are extracted through the convolution process. Because 3D CNNs account inherently for the spatial relationships embedded within the data, we test the hypothesis that CNN models have improved predictions over the ML models because of their ability to learn the relationships between 3D microstructural arrangements and corresponding mechanical properties. A secondary objective of this work is to quantify the relative improvement of the CNN-model predictions when the training data include progressively informative features beyond just grain IDs. Based on the ML- and DL-model predictions of effective yield strength, the following conclusions are made:

- The simple Ridge regression model, essentially a linear least-squares model with L2 regularization, performed approximately the same as the more sophisticated XGBoost model. Both models captured trends and patterns visible in the reference maps; however, the spatial variability was generally underpredicted.
- The CNN models that used grain IDs as the primary input generally had the worst predictions among all of the ML and DL models considered; whereas, the CNNs that used quaternions as input had the best performance among all of the models. The CNN model trained using only quaternions, without any other information, achieved an R^2 value of 0.84 and an RMSE value of 16.57 MPa based on blind predictions of holdout layers in two different MAM build domains. This confirms that, unlike grain IDs, crystal orientations provide a physically based input from which relevant information about crystallographic arrangements and textures can be extracted and linked to mechanical properties. Based on this conclusion, the posed hypothesis is conditionally accepted.
- Incorporating progressively informative auxiliary features as input to the CNN models (using either grain IDs or quaternions as the primary input) resulted in a slight improvement with the global loading vector (R^2 increase between 0.02 and 0.06), either no change or worse performance with the Schmid factor (R^2 change between 0.0 and -0.03), and a significant improvement with the micromechanical Taylor factor (R^2 improvement between 0.10 and 0.26).
- The computational time required to predict the entire map of effective yield strength for one layer in a single build domain was approximately 85 CPU hours using the elasto-viscoplastic fast Fourier transform model (i.e., the reference solution) and less than two seconds on a single processor using Ridge regression, XGBoost, or CNN models.

The results from this work demonstrate that, once suitably trained, certain data-driven models could complement physics-driven modeling by massively expediting structure-property predictions, which could enable high-throughput predictions of property maps and rapid screening of MAM builds based on specified property tolerances. Such capabilities could have significant implications for MAM design, optimization, and qualification. Among all of the data-driven models tested here, CNN models that use crystal orientation as input provide the best overall predictions and have the added benefit of requiring little to no pre-processing for feature extraction.

Data availability

The raw/processed data required to reproduce these findings cannot be shared at this time as the data also forms part of an ongoing study.

CRediT authorship contribution statement

Carl Herriott: Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing - original draft. **Ashley D. Spear:** Conceptualization, Resources, Writing - review & editing, Supervision, Funding acquisition.

Acknowledgement

The authors wish to thank Dr. Wenda Tan and Xuxiao Li for their role in previous work that provided the foundation for this study. Dr. Aowabin Rahman is gratefully acknowledged for fruitful discussions that influenced aspects of this research. This material is based upon work supported by the U.S. Department of Defense Office of Economic Adjustment under award No. ST1605-19-03. The support and resources from the Center for High Performance Computing at the University of Utah are gratefully acknowledged.

References

[1] A. Ahmadi, R. Mirzaeifar, N.S. Moghaddam, A.S. Turabi, H.E. Karaca, M. Elahinia, Effect of manufacturing parameters on mechanical properties of 316l stainless steel parts fabricated by selective laser melting: a computational framework, Mater. Des. 112 (2016) 328–338.
[2] M.T. Andani, M.R. Karamooz-Ravari, R. Mirzaeifar, J. Ni, Micromechanics modeling of metallic alloys 3d printed by selective laser melting, Mater. Des. 137 (2018) 204–213.
[3] W. Yan, Y. Lian, C. Yu, O.L. Kafka, Z. Liu, W.K. Liu, G.J. Wagner, An integrated process-structure-property modeling framework for additive manufacturing, Comput. Methods Appl. Mech. Eng. 339 (2018) 184–204.
[4] C. Herriott, X. Li, N. Kouraytem, V. Tari, W. Tan, B. Anglin, A.D. Rollett, A.D. Spear, A multi-scale, multi-physics modeling framework to predict spatial variation of properties in additive-manufactured metals, Modell. Simul. Mater. Sci. Eng. 27 (2019) 025009.
[5] T. Ozturk, A.D. Rollett, Effect of microstructure on the elasto-viscoplastic deformation of dual phase titanium structures, Comput. Mech. 61 (2018) 55–70.
[6] R.A. Lebensohn, A.K. Kanjarla, P. Eisenlohr, An elasto-viscoplastic formulation based on fast fourier transforms for the prediction of micromechanical fields in polycrystalline materials, Int. J. Plast 32–33 (2012) 59–69.
[7] V. Tari, R.A. Lebensohn, R. Pokharel, T.J. Turner, P.A. Shade, J.V. Bernier, A.D. Rollett, Validation of micro-mechanical FFT-based simulations using high energy diffraction microscopy on Ti-7Al, Acta Mater. 154 (2018) 273–283.
[8] A.D. Spear, S.R. Kalidindi, B. Meredig, A. Kontsos, J.-B. le Graverend, Data-driven materials investigations: the next frontier in understanding and predicting fatigue behavior, JOM 70 (2018) 1143–1146.
[9] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (2015) 436.
[10] J. Schmidhuber, Deep learning in neural networks: an overview, Neural Networks 61 (2015) 85–117.
[11] A. Cecen, H. Dai, Y.C. Yabansu, S.R. Kalidindi, L. Song, Material structure-property linkages using three-dimensional convolutional neural networks, Acta Mater. 146 (2018) 76–84.
[12] J. Jung, J.I. Yoon, H.K. Park, J.Y. Kim, H.S. Kim, An efficient machine learning approach to establish structure-property linkages, Comput. Mater. Sci. 156 (2019) 17–25.
[13] A.L. Frankel, R.E. Jones, C. Alleman, J.A. Templeton, Predicting the mechanical response of oligocrystals with deep learning, 2019, arXiv preprint arXiv:1901.10669.
[14] A. Mangal, Applied Machine Learning to Predict Stress Hotspots in Materials (Ph.D. thesis), Carnegie Mellon University, 2018. Copyright – Database copyright ProQuest LLC; ProQuest does not claim copyright in the individual underlying works; Last updated – 2018-08-24.
[15] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 3431–3440.
[16] A. Bansal, X. Chen, B. Russell, A. Gupta, D. Ramanan, Pixelnet: Representation of the pixels, by the pixels, and for the pixels, 2017. arXiv preprint arXiv:1702.06506.
[17] C. Szegedy, S. Ioffe, V. Vanhoucke, A.A. Alemi, Inception-v4, inception-resnet and the impact of residual connections on learning, in: Thirty-First AAAI Conference on Artificial Intelligence.
[18] D. Raabe, M. Sachtleber, Z. Zhao, F. Roters, S. Zaefferer, Micromechanical and macromechanical effects in grain scale polycrystal plasticity experimentation and simulation, Acta Mater. 49 (2001) 3433–3441.
[19] X. Li, W. Tan, Numerical investigation of effects of nucleation mechanisms on grain

structure in metal additive manufacturing, Comput. Mater. Sci. 153 (2018) 159–169.

[20] C.A. Gandin, J.L. Desbiolles, M. Rappaz, P. Thevoz, A three-dimensional cellular automation-finite element model for the prediction of solidification grain structures, Metall. Mater. Trans. A 30 (1999) 3153–3165.

[21] W. Tan, Y.C. Shin, Multi-scale modeling of solidification and microstructure development in laser keyhole welding process for austenitic stainless steel, Comput. Mater. Sci. 98 (2015) 446–458.

[22] M.A. Groeber, M.A. Jackson, Dream.3D: a digital representation environment for the analysis of microstructure in 3D, Integrating Mater. Manuf. Innov. 3 (2014) 5.

[23] J. Luster, M.A. Morris, Compatibility of deformation in two-phase ti-al alloys: dependence on microstructure and orientation relationships, Metall. Mater. Trans. A 26 (1995) 1745–1756.

[24] D. Mainprice, F. Bachmann, R. Hielscher, H. Schaeben, Descriptive tools for the analysis of texture projects with large datasets using MTEX: strength, symmetry and components, Geol. Soc., London, Special Publications 409 (2014) 251–271.

[25] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: machine learning in python, J. Mach. Learn. Res. 12 (2011) 2825–2830.

[26] A.E. Hoerl, R.W. Kennard, Ridge regression: biased estimation for nonorthogonal problems, Technometrics 12 (1970) 55–67.

[27] T. Chen, C. Guestrin, Xgboost: a scalable tree boosting system, CoRR abs/1603.02754, 2016.

[28] D. Nielsen, Tree Boosting With XGBoost-Why Does XGBoost Win Every Machine Learning Competition?, Master's thesis, NTNU, 2016.

[29] J. Bergstra, Y. Bengio, Random search for hyper-parameter optimization, J. Mach. Learn. Res. 13 (2012) 281–305.

[30] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: International Conference on Learning Representations.

[31] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, CoRR abs/1512.03385, 2015.

[32] A.F. Agarap, Deep learning using rectified linear units (relu), CoRR abs/1803.08375, 2018.

[33] G.E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Improving neural networks by preventing co-adaptation of feature detectors, CoRR abs/1207.0580, 2012.

[34] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, in: 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7–9, 2015, Conference Track Proceedings.

[35] S.J. Reddi, S. Kale, S. Kumar, On the convergence of adam and beyond, International Conference for Learning Representations abs/1904.09237, 2019.

[36] B.J. Hayes, B.W. Martin, B. Welk, S.J. Kuhr, T.K. Ales, D.A. Brice, I. Ghamarian, A.H. Baker, C.V. Haden, D.G. Harlow, et al., Predicting tensile properties of ti-6al-4v produced via directed energy deposition, Acta Mater. 133 (2017) 120–133.