A Deep Learning Approach for Polycrystalline Microstructure-Statistical Property Prediction

José Pablo Quesada-Molina[1,2] [0000-0002-3358-284X] and Stefano Mariani[1] [0000-0001-5111-9800]

[1] Department of Civil and Environmental Engineering, Politecnico di Milano, Piazza Leonardo da Vinci 32, 20133, Milano (Italy)
[2] Department of Mechanical Engineering, University of Costa Rica, San Pedro Montes de Oca, San José (Costa Rica)

Abstract. Upscaling of the mechanical properties of polycrystalline aggregates might require complex and time-consuming procedures, if adopted to help in the design and reliability analysis of micro-devices. In inertial micro electro-mechanical systems (MEMS), the movable parts are often made of polycrystalline silicon films and, due to the current trend towards further miniaturization, their mechanical properties must be characterized not only in terms of average values but also in terms of their scattering. In this work, we propose two convolutional network models based on the ResNet and DenseNet architectures, to learn the features of the microstructural morphology and allow automatic upscaling of the statistical properties of the said film properties. Results are shown for film samples featuring different values of a length scale ratio, so as to assess accuracy and computational efficiency of the proposed approach.

Keywords: Multi-scale analyses, homogenization, scattered mechanical properties, deep learning, convolutional neural networks, ResNet, DenseNet.

1 Introduction

Design and reliability of micro-devices like MEMS rely more and more on digital twins, that are numerical models of their mechanical parts (but also of their electronics, though this is not of concern here) allowing for all the possible epistemic uncertainties [1-4]. Even if uncertainties in MEMS readout are not targeted as a great issue, since different methods exist to compensate for them [5], it must be stated that they can overall lead to a detrimental effect on the relevant device performance indices [6,7].

The ever-increasing need for downsizing the devices, on top of all for economic reasons, tends to enhance issues linked to the scattering in the measured response of the devices to the external actions, in most of the cases induced by a high sensitivity to micromechanical features and defects. For polysilicon microstructures, in recent works we showed how the morphology of the films and defects caused by the microfabrication process, like e.g. overetch, can be properly accounted for in stochastic analyses to cover the aforementioned scattered experimental data [8-12]. The price to attain such an accuracy of models handled in Monte Carlo simulations, is that the statistics of the sought mechanical properties have to be computed every time being scale dependent, namely

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77977-1_44

being affected by the interaction between the length scale describing the movable structure of the device and the length scale describing the film morphology (for instance, proportional to a characteristic radius of the grains).

The mentioned length scale separation is very important for computing the homogenized properties of the polysilicon films in multi-scale and, often, multi-physics analyses [13,14]. To avoid any surrogate or smoothing procedure of the available results, upscaling through a new Monte Carlo analysis thus looks necessary whenever the considered value does not match those already investigated. Starting in [15,16], we proposed a deep learning approach to this problem, aiming to learn the micromechanical features of the polycrystal and their role in setting the overall property of interest, see also [17-22], as anticipated not only in terms of reference values but also in terms of scattering or, in a general sense, of its probability distribution. We started by dealing with the elastic moduli of the considered textured films, and showed the importance of the *quality* of the dataset of morphology pictures used to train a convolutional network within an image recognition-like approach.

In this work, we move on by assessing the performances of different network architectures to foresee an optimization of the entire procedure, in terms of accuracy of the results, generalization capability (to catch all the length scale separation effects) and computational efficiency. The formerly used ResNet architecture [23] is here compared to the newly proposed DenseNet one [24]. Results show that both have distinctive features and can attain a remarkable accuracy, higher in the case of the DenseNet-based model.

The remainder of this paper is organized as follows. In Section 2, the considered scattering in the micromechanically-driven response of the polycrystalline films is discussed, and the samples used to generate the datasets are accordingly defined. Section 3 deals with the proposed methodology to specify data generation and pre-processing, the features of the adopted convolutional network-based modes and their expected effects on the outcomes for the ResNet and DenseNet architectures. Results are reported in Section 4 for polysilicon films characterized by different values of the length scale separation parameter in the training and test sets, to assess the already mentioned generalization capability of the approach. Some concluding remarks and proposals for future developments are finally gathered in Section 5.

2 Statistical Scatter in the Homogenized Polysilicon Properties at the Mesoscale

In compliance with a standard geometry of MEMS movable structures, the polysilicon microstructures to be characterized feature a rather small ratio between the size L of the polycrystalline aggregate (termed mesoscale and representing the scale over which homogenization is being carried out) and the characteristic size d of the microscopic heterogeneities (termed microscale and representing the average size of silicon grains).

Whenever a homogenization procedure is adopted within such a frame to define the overall mechanical, electromagnetic, thermal, or any other type of properties in a het-

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77977-1_44

erogeneous medium, the mesoscale properties themselves are termed *apparent*. *Effective* properties instead refer to samples of the heterogeneous material large enough to comply with a kind of asymptotic rule of mixture to hold. In this latter case, the material domain can be regarded as mesoscopically homogeneous and L defines the size of the corresponding Representative Volume Element (RVE), see e.g. [25-27] for polycrystalline situations related to the specific case considered in this study. Through homogenization, the overall properties of interest are supposed to be realization independent, namely independent of the micromechanical features of the polycrystal. As said, this condition is achieved only asymptotically, when $\delta = L/d \to \infty$ or, at least, when it becomes so large that grain effects lead to marginal variations in the measured response. Such a critical threshold is attained for δ on the order of $10 - 100$, depending on the property to be determined.

As reported in [27], for polysilicon a microstructure featuring $\delta \cong 60$ could be regarded as an RVE, while domains featuring smaller values do not to comply with the asymptotic values provided by standard homogenization procedures and therefore require the scattering in the results to be assessed too. Under such conditions, Statistical Volume Elements (SVEs) are instead adopted to represent the polycrystalline material samples; since each single realization can be representative of the material only in a statistical sense, Monte Carlo simulations are necessary to quantify a mean property value and also the scattering of the results around it.

3 Methodology

3.1 Input Data Generation and Pre-Processing

The polycrystalline morphology here considered is an epitaxially grown one. Moving from the free surface of the substrate, each grain has a major crystal orientation almost aligned with the direction perpendicular to the surface itself. By disregarding the scattering in such a texture, in our former works we focused on a thin slice of the polycrystal and therefore simplify the geometry allowing for two-dimensional SVEs. Such a simplification can be classified as a kind of dimensionality-reduction for the problem at hand.

A set of SVEs has been digitally generated via Voronoi tessellations. Each SVE features its own grain boundary network geometry, and orientations of the crystal lattice of all the grains gathered. The Monte Carlo procedure, exploited to quantify the stochastic effects due to the grain arrangements on the probability distribution of the apparent Young's Modulus E, has been then fed by the results of a numerical homogenization carried out for all the SVEs. The obtained results represent the ground-truth data (or the labels) to be used during the training of the neural networks (NNs). Figure 1 provides an illustration of a (not-to-scale) couple of characteristic morphologies of the handled SVEs, respectively featuring a 2μm×2μm (left) or 5μm×5μm (right) size. In these single-channel images, the color encodes the in-plane lattice orientation θ of each single-crystalline domain or grain, measured relative to a global reference axis (e.g. the horizontal one in the figure).

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77977-1_44

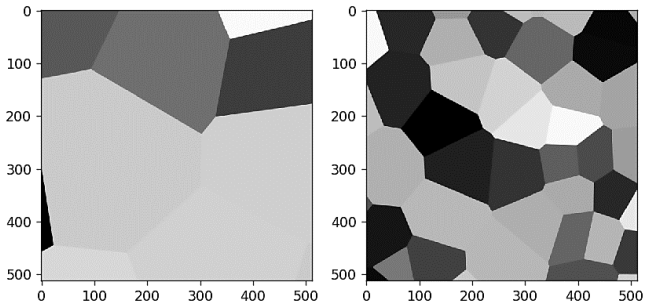

Fig. 1. Digitally generated two-dimensional polysilicon microstructures, wherein the pixel coordinates are represented along the SVE axes.

The artificially generated images have been then pre-processed in order to maximize the available distinctive features and reduce artifacts, represented by pixels with incorrect values assigned along the grain boundaries where grains with different orientation merge. A median filter with kernel size [3,3] has been adopted, to smooth the input images and therefore reduce the mentioned artifacts. To also reduce the computational cost of training, the images have been resized to have a final resolution of 128×128 pixels, moving from the original 512×512 pixels one. Additional details regarding the generation of the input data, as well as the pre-processing steps, can be found in [16]. Figure 2 provides a sketch of the final resolution for the two microstructures gathered in Figure 1.

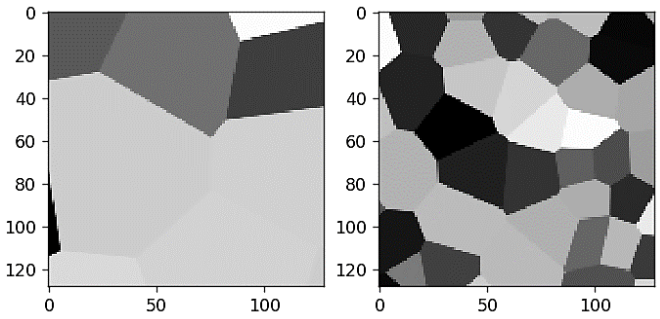

Fig. 2. Final resolution for the SVEs of Figure 1, wherein pixel coordinates are again represented along the SVE axes.

3.2 Proposed Models and Implementation

Results discussed in [15,16] are taken as the baseline in the current research activity. Aiming to exploit the advantages of the convolutional architecture proposed in [24], a densely connected network has been newly adopted as the specific architecture to carry on feature extraction, that is the key stage upon which the overall performance of the

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77977-1_44

regression model is based. DenseNet121 is proposed to be employed for feature learning, and its performance is going to be directly compared to that of the former model based instead in the use of the residual ResNet18. Additionally, a new test set featuring SVEs with size 3μm×3μm has been generated and allowed for in the analysis, in order to further assess the generalization capabilities of both models regarding intermediate length scales.

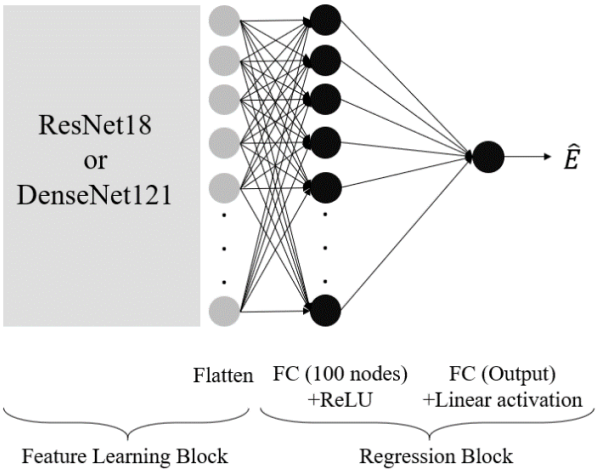

Fig. 3. Pictorial description of the adopted models: the difference between them is represented by the convolutional architecture employed as the Feature Learning Block (either the ResNet18 or the DenseNet121).

From a methodological standpoint, once each microstructural representation has been generated and preprocessed, the relevant image has been individually fed to the specific NN under consideration. With this input information, each NN model consists of an initial stage aimed at feature extraction taking place in the *feature learning block*; this activity is done thanks to the use of a convolutional network architecture, either the ResNet18 baseline or the newly proposed DenseNet121. After feature extraction, the output feature maps on the last convolutional layer undergo a flatten operation, after which the high-level features extracted are employed as input of the *regression block*, made of a standard fully connected multilayer perceptron. The aim of this block is to provide the estimation of the effective property of interest, associated to the microstructure. For comparison convenience, it is important to mention that all the design elements associated to the regression block have been kept exactly the same for both models, in order to highlight the effect of the convolutional architecture on the overall prediction and generalization capabilities of the model. Besides the regression blocks coupled on top of feature learning, also all the associated hyperparameters but the mini-batch size, which has been set to be the maximum number of computationally manageable samples, have been kept the same for both models. The regression block is characterized by an arrangement of a dense fully connected layer (100 nodes) + ReLU activation, followed by a fully connected output layer (1 node) + Linear activation. The

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77977-1_44

total number of parameters in the ResNet18-based regression model has turned out to be 14,457,612 while the total number of parameters in the DenseNet121-based regression model has been 8,669,833. An illustration of the architecture of the two models is presented in Figure 3: note that the output corresponds to the apparent Young's modulus prediction for each image, that is to $\hat{E}$.

Hence, the aim of the first stage of training has been to learn the individual mapping of the input data onto the effective SVE property, that is dealt with as a kind of label. After training, the statistical analysis of the estimated $\hat{E}$ values and the evaluation of the mean value and dispersion around it are readily accomplished from the set of mappings, by simply extracting the relevant statistical indicators of interest. Therefore, during the training stage a certain number of microstructures have been employed to learn the underlying mapping between the topology of the grain boundary network and the lattice orientation of each grain on one side, and the overall elastic properties of a polycrystalline aggregate on the other side. After this stage, a testing stage has followed, in which each regression model has been used to predict the value of the Young's modulus over new, unseen SVEs featuring also length scales δ different from those used during the training.

For this stage of the procedure, early stopping has been implemented as the regularization technique by monitoring the validation loss, with the patience parameter set to 50 epochs. The weights obtained at the end of the learning stage and corresponding to the minimum value of the validation loss, have been later adopted in order to assess the performance of the models on the test set. Table 1 summarizes the main hyperparameters selected.

Results to follow have been obtained with the two models developed making use of the Keras API, based on TensorFlow. To speed up training and testing, a GeForce RTX 2080 GPU has been exploited.

Table 1. Hyperparameters used during the training of both models.

Hyperparameter	Value
Total number of epochs	500
Patience (early stopping)	50
Mini-Batch Size (B.S)	300[1], 85[2]
Learning rate α	$5 \cdot 10^{-4}$
Optimizer	Adam
Loss function	MSE[3]

[1] Maximum number of computationally manageable samples for the ResNet-based model.
[2] Maximum number of computationally manageable samples for the DenseNet-based model.
[3] Mean squared error.

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77977-1_44

3.3 ResNet vs DenseNet: Addition vs Concatenation of Features

Considering the aspects of the implementation mentioned in Section 3.2, it is expected to note performance differences between the models due to the specific connectivity pattern displayed by the two different architectures of ResNets and DenseNets.

According to the theoretical and practical evidence presented in [24], the connectivity pattern in DenseNets is characterized by all layers with matching feature-map sizes directly connected with each other. As explained in the original work, feature propagation and reuse is therefore by definition strengthened in this convolutional network architecture, to improve the flow of information and gradients throughout the network and substantially reduce the number of parameters to tune during training. This last aspect is linked to the relative *narrow* nature of DenseNets: each non-linear transformation at every layer adds just a few feature-maps to the network *collective knowledge*. The number of filters per layer, referred to by the authors of this architecture as growth rate k, is low if compared to the number of filters typically used in ResNet architectures: in concrete terms, it is *32* for the DenseNet121 model, while it amounts to 64, 128, 256 and 512 for the ResNet18 model.

Denoting the output of the l-th NN layer as x_l and the nonlinear transformation as $H_l(\cdot)$, with traditional convolutional networks, ResNet and DenseNet we respectively have:

$$x_l = H_l(x_{l-1}) \tag{1}$$

$$x_l = H_l(x_{l-1}) + x_{l-1} \tag{2}$$

$$x_l = H_l([x_0, x_1, \ldots, x_{l-1}]) \tag{3}$$

where $[x_0, x_1, \ldots, x_{l-1}]$ refers to the concatenation of the feature-maps produced in layers 0, 1, …, $l-1$. In these equations, $H_l(\cdot)$ is a composite function of consecutive operations. For example, in both ResNets and DenseNets the nonlinear transformation includes the consecutive application of three operations: a batch normalization, a rectified linear unit and a convolution.

At variance with ResNets, features in DenseNets are not combined through summation before they are passed into a layer (symbolized in Eq. 2); instead, they are combined by concatenation (symbolized in Eq. 3). This difference largely determines the characteristics of the information extraction and propagation in each model, ultimately constituting the key difference when comparing the performances of the two models here considered.

4 Results

4.1 Generalization Capability of Trained Models

The data to be handled by the two models have been split exactly in the same manner. A total of 3878 2μm×2μm images (δ=2μm/0.5μm=4) has been split into the train-

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77977-1_44

ing set and validation sets, using ~75% and 25% respectively. In this way, the parameters of the model are initially fit on the training set, while simultaneously, after each epoch, the fitted model is used to predict the response over the validation set. The validation set therefore has a double purpose: it is used to tune the hyperparameters of the model and it allows for the implementation of *early stopping* as a regularization technique. Finally, two additional sets referred to as test sets have been generated using 3μm×3μm images and 5μm×5μm images. Clearly, although these two sets feature different δ values, specifically δ=3μm/0.5μm=6 and δ=5μm/0.5μm=10, both preserve the input size i.e. image resolution. These sets are employed for the exclusive purpose of assessing the performance (i.e. generalization) of the final model. Summarizing, the datasets are arranged as follows: 2889 2μm×2μm images as training set (Mean = 150.08 GPa and SD = 5.40 GPa); 989 2μm×2μm images as validation set (Mean = 150.16 GPa and SD = 5.34 GPa), and, as alternate test sets featuring a larger aggregate size, 30 3μm×3μm images (Mean = 149.87 GPa and SD= 3.75 GPa) and 145 5μm×5μm images (Mean = 149.34 GPa and a SD = 2.44 GPa) were generated. Within the brackets, for each set values have been reported for the relevant mean and standard deviation (SD) of E.

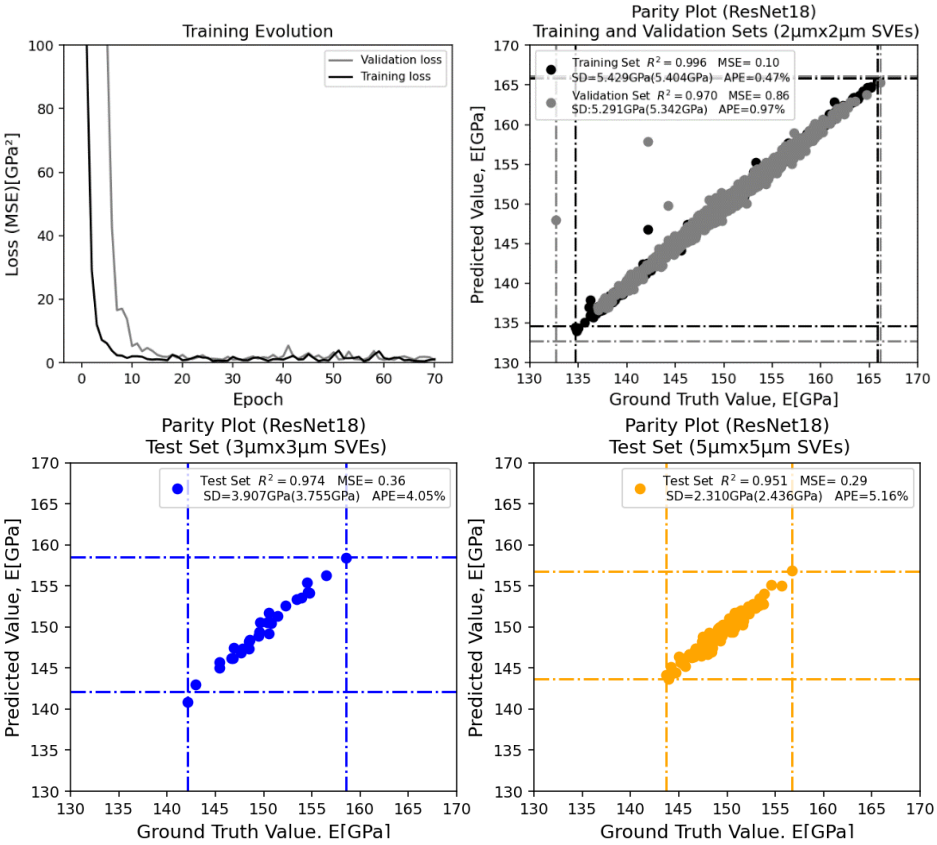

Fig. 4. Results of the ResNet-based regression model.

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77977-1_44

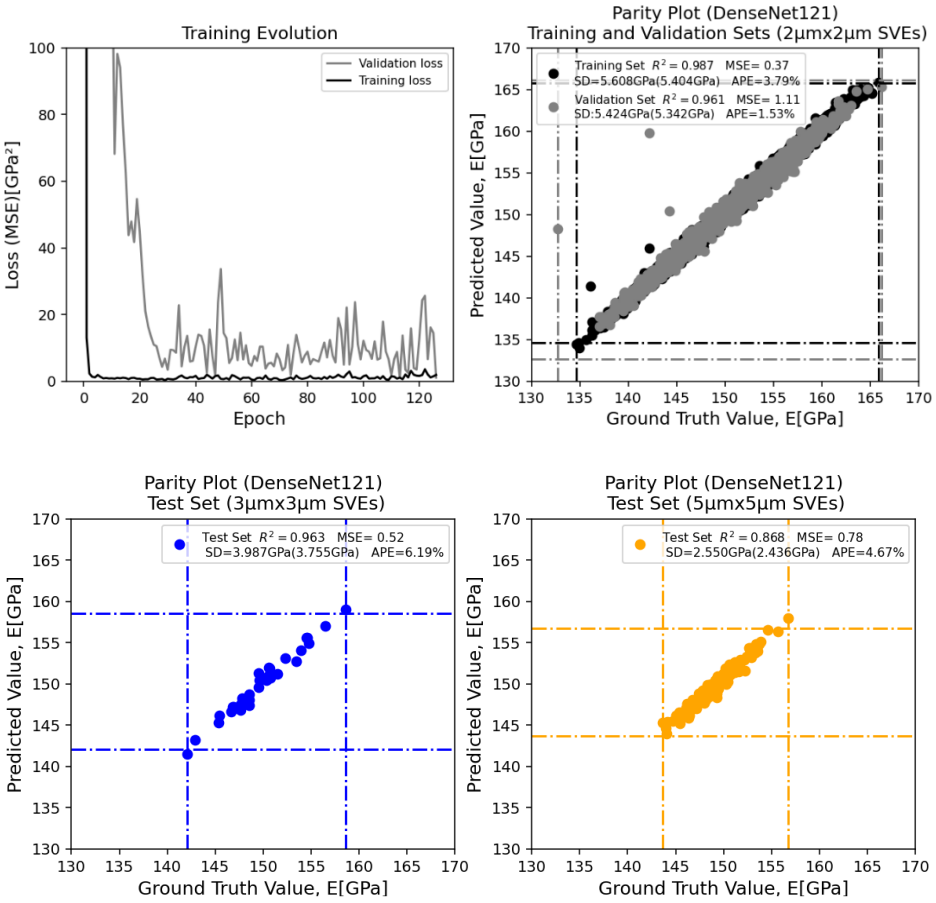

Fig. 5. Results of the DenseNet-based regression model.

Figures 4 and 5 show the obtained results with the ResNet-based and the DenseNet-based regression models, respectively. For each case, results are reported first in terms of the evolution of the training and validation loss against the epochs. To assess the performance of the models, results are reported next in terms of parity plots. In these charts, the dotted lines are used to denote the maximum and minimum label values featured by each dataset. In this way, predictions should ideally be mapped within the limits defined by these lines (i.e. inside the dotted squares). Moreover, for an optimal trained model, the data should map the identity function for every dataset, with all the dots aligned along the 45° diagonal. The performance of the regression models is assessed through the coefficient of determination, R^2 and the MSE. In the best case, e.g. predicted values exactly match the ground-truth values, then $R^2 = 1$ and MSE = 0. For each dataset, the mentioned evaluation metrics are reported in the legends. The legends also include the estimation of the statistical indicator of interest (SD); these values are computed from the set of individual predictions produced by the trained model. The corresponding SD characterizing the ground-truth data appears within the brackets and

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77977-1_44

the associated absolute percentage error (APE) is finally reported. Results are summarized in Table 2.

Table 2. Summary of the results

Metric	Dataset	ResNet18-based	DenseNet121-based
MSE [GPa^2]	Training set	0.1	0.37
	Validation set	0.86	1.11
	Test set 1 (δ=6)	0.36	0.52
	Test set 2 (δ=10)	0.29	0.78
R^2	Training set	0.996	0.987
	Validation set	0.970	0.961
	Test set 1 (δ=6)	0.974	0.963
	Test set 2 (δ=10)	0.951	0.868
Standard Deviation APE	Training set	0.47%	3.79%
	Validation set	0.97%	1.53%
	Test set 1 (δ=6)	4.05%	6.19%
	Test set 2 (δ=10)	5.16%	4.67%

For the case of the ResNet18-based model in Figure 4, the learning process reached a plateau after 71 epochs, with the subsequent 50 epochs leading to small validation loss changes. This model thus shows promising generalization capabilities in setting the intended link between the microstructure and the investigated property, as confirmed by the results provided for both test sets. For the for the test dataset δ=6, the model has effectively predicted a larger value for the statistical indicator describing the dispersion of the homogenized property, as expected for the lower value of δ associated to this test set, when compared to the one featuring δ=10.

In the case of the DenseNet121-based model in Figure 5, the learning process reached a plateau after 126 epochs, again with the subsequent 50 epochs not leading to clear changes in the validation loss. This new model, which is linked to a lower total number of parameters to be tuned in the training process, has not displayed significant performance improvements, when compared to the ResNet18-based.

In addition, by analyzing the presented results, it can be observed that although the DenseNet121-based model requires fewer parameters it does not necessarily display a faster convergence rate, e.g. in terms of the number of epochs required to reach the reported minimum validation loss values. Moreover, the computational burden of this model, in terms of memory usage and GPU power, has been higher. Individual epoch of the DenseNet121-based model lasted about 3 times the time taken by epochs in the ResNet18-based. Furthermore, due to memory issues, the DenseNet121-based model has only allowed the use of a fraction of the mini batch size when compared to the ResNet18-based model (as stated in Table 1), leading to a noisier convergence behavior as can be observed in the training evolution plots. This is interesting, since the lower number of trainable parameters associated to the DenseNet121-based model does not

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77977-1_44

proportionally translate into larger number of computationally manageable samples in the batch size, or faster training time.

As far as the generalization capabilities are concerned, in general terms a better performance has been observed adopting the ResNet18-based architecture.

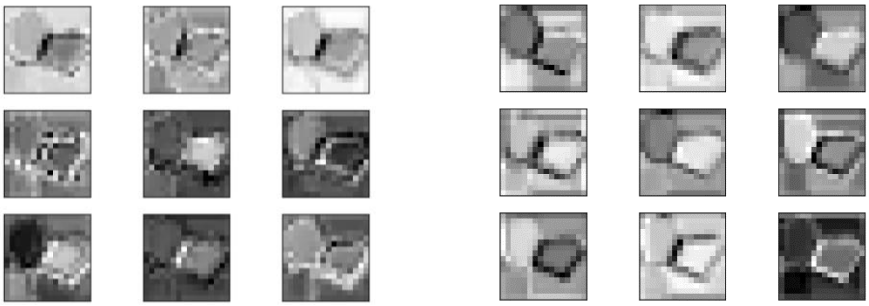

Fig. 6. Random output feature maps. ResNet18 (left) and the DenseNet121 (right).

Due to intrinsic differences between both convolutional network architectures, DenseNet121-based model is able to extract higher level features: while for the ResNet18, the output size of the layer just before the flatten operation is [8×8×512], for the DenseNet121 this becomes [4×4×1024], featuring a larger down sampling and number of output features maps (OFMs). As an additional remark, Figure 6 shows a comparison between the information extracted in a group of randomly selected, intermediate, same size OFMs, once both models have been already trained. A qualitative comparison reveals that the DenseNet-based feature extraction better reproduces the morphology and internal homogeneity within the grains: the distribution of pixels with incorrect values along grain boundaries, observed in the previous figure in the form of pixelated fringes, appears to be thinner for the OFMs provided by the DenseNet121.

5 Conclusions

In this work, we have compared the performances of two different NN-based models to learn the micromechanical features ruling the value of the apparent stiffness of a polycrystalline aggregate. Specifically, for SVEs characterized by a mesoscale size slightly larger than the micromechanical length scale, linked e.g. to the characteristic size of the grains, the overall Young's modulus of textured polycrystalline films has been estimated in statistical terms through its mean value (though not discussed here as not significant in this work) and scattering around it.

For the feature learning stage of the models, the two different architectures adopted have been the ResNet18 and the DenseNet121. For the problem at hand, the first one has turned out to be characterized by a larger set of parameters to be tuned during the

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77977-1_44

training of the NN; the second one, though featuring less parameters to tune, has resulted to be more computational expensive. Moreover, the additional computational burden associated to the second architecture did not result into higher accuracy in mapping the micromechanical features ruling the results, and so, in catching the scattering in the results attained with numerical homogenization procedures, when compared to the first one.

The proposed approach is still in its infancy and surely requires some additional work in order to optimize the proposed physics-informed NN architecture, the setting of the NN hyperparameters, and the training strategy. The goal of the project is to make this procedure available for all the devices whose response may be sensitive to microstructural features and, accordingly, to possible defects existing in their building blocks at the nano- and microscale.

Acknowledgments

Partial financial support by STMicroelectronics through the project *MaRe* (Material Reliability) is gratefully acknowledged. JPQM acknowledges the financial support by the University of Costa Rica.

References

1. Hsu, T.R.: MEMS and Microsystems: Design, Manufacture, and Nanoscale Engineering. John Wiley & Sons, Hoboken, NJ, USA, (2008)
2. Brand, O., Fedder, G.K., Hierold, C., Korvink, J.G., Tabata, O., Tsuchiya, T.: Reliability of MEMS: Testing of Materials and Devices. JohnWiley & Sons, Hoboken, NJ, USA, (2013)
3. Corigliano, A., Ardito, R., Comi, C., Frangi, A., Ghisi, A., Mariani, S.: Mechanics of Microsystems. John Wiley & Sons, Hoboken, NJ, USA, (2018)
4. Weinberg, M.S., Kourepenis, A.: Error sources in in-plane silicon tuning-fork MEMS gyroscopes. Journal of Microelectromechanical Systems. **15**, 479–491 (2006). doi: 10.1109/JMEMS.2006.876779
5. De Laat, M., Pérez Garza, H., Herder, J., Ghatkesar, M.: A review on in situ stiffness adjustment methods in MEMS. Journal of Micromechanics and Microengineering. **26**, 1-21 (2016). doi: 10.1088/0960-1317/26/6/063001
6. Uhl, T., Martowicz, A., Codreanu, I., Klepka, A.: Analysis of uncertainties in MEMS and their influence on dynamic properties. Archives of Mechanics. **61**, 349–370 (2009).
7. Bagherinia, M., Mariani, S.: Stochastic Effects on the Dynamics of the Resonant Structure of a Lorentz Force MEMS Magnetometer. Actuators. **8**, 36 (2019). doi: 10.3390/act8020036
8. Mirzazadeh, R., Mariani, S.: Uncertainty quantification of microstructure-governed properties of polysilicon MEMS. Micromachines. 8, 248 (2017). doi: 10.3390/mi8080248
9. Mirzazadeh, R., Ghisi, A., Mariani, S.: Statistical Investigation of the Mechanical and Geometrical Properties of Polysilicon Films through On-Chip Tests. Micromachines. **9**, 53 (2018). doi: 10.3390/mi9020053
10. Mariani, S., Ghisi, A., Mirzazadeh, R., Eftekhar Azam, S.: On-Chip Testing: A Miniaturized Lab to Assess Sub-Micron Uncertainties in Polysilicon MEMS. Micro and Nanosystems. **10**, 84-93 (2018). doi: 10.2174/1876402911666181204122855

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77977-1_44

11. Mirzazadeh, R., Eftekhar Azam, S., Mariani, S.: Mechanical Characterization of Polysilicon MEMS: A Hybrid TMCMC/POD-Kriging Approach. Sensors. **18**, 1243 (2018). doi: 10.3390/s18041243
12. Ghisi, A., Mariani, S.: Effect of imperfections due to material heterogeneity on the offset of polysilicon MEMS structures. Sensors. **19**, 3256 (2019). doi: 10.3390/s19153256
13. Mariani, S., Ghisi, A., Corigliano, A., Martini, R., Simoni, B.: Two-scale simulation of drop-induced failure of polysilicon MEMS sensors. Sensors. **11**, 4972-4989 (2011). doi: 10.3390/s110504972
14. Ghisi, A., Mariani, S., Corigliano, A., Zerbini, S.: Physically-based reduced order modelling of a uni-axial polysilicon MEMS accelerometer. Sensors. **12**, 13985-14003 (2012). doi: 10.3390/s121013985
15. Quesada-Molina, J.P., Rosafalco, L., Mariani, S.: Stochastic mechanical characterization of polysilicon MEMS: a Deep Learning approach. In: 6th International Electronic Conference on Sensors and Applications. Proceedings. **42**, 8 (2020). doi: 10.3390/ecsa-6-06574
16. Quesada-Molina, J.P., Rosafalco, L., Mariani, S.: Mechanical characterization of polysilicon MEMS devices: a stochastic, Deep Learning-based approach. In: 2020 21st International Conference on Thermal, Mechanical and Multi-Physics Simulation and Experiments in Microelectronics and Microsystems (EuroSimE), pp. 1-8, IEEE Press, New York (2020). doi: 10.1109/EuroSimE48426.2020.9152690.
17. Bock, F.E., Aydin, R.C., Cyron, C.J., Huber, N., Kalidindi, S.R., Klusemann, B.: A Review of the Application of Machine Learning and Data Mining Approaches in Continuum Materials Mechanics. Frontiers in Materials. **6**, 110 (2019). doi: 10.3389/fmats.2019.00110
18. Cang, R., Ren, M.Y.: Deep Network-Based Feature Extraction and Reconstruction of Complex Material Microstructures. In: Proc. of the ASME 2016 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference, pp. 1-10 (2016). doi: 10.1115/DETC2016-59404
19. Lubbers, N., Lookman, T., Barros, K.: Inferring low-dimensional microstructure representations using convolutional neural networks. Physical Review E. **96**, 1-14 (2017). doi: 10.1103/PhysRevE.96.052111
20. Yang, Z., Yabansu, Y. C., Al-Bahrani, R., Liao, W.-k., Choudhary, A. N., Kalidindi, S. R. Agrawal, A.: Deep learning approaches for mining structure-property linkages in high contrast composites from simulation datasets. Computational Materials Science. **151,** 278-287 (2018). doi: 10.1016/j.commatsci.2018.05.014
21. Cecen, A., Dai, H., Yabansu, Y.C., Kalidindi, S.R., Song, L.: Material structure-property linkages using three-dimensional convolutional neural networks. Acta Materialia. **146**, 76-84 (2018). doi: 10.1016/j.actamat.2017.11.053
22. Cang, R., Li, H., Yao, H., Jiao, Y., Ren, Y.: Improving direct physical properties prediction of heterogeneous materials from imaging data via convolutional neural network and a morphology-aware generative model. Computational Materials Science. **150**, 212-221 (2018). doi: 10.1016/j.commatsci.2018.03.074
23. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proc. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770-778, IEEE Press, New York (2016). doi: 10.1109/CVPR.2016.90
24. Huang, G., Liu, Z., Van der Maaten, L., Weinberger,K.Q.: Densely Connected Convolutional Networks. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2261-2269, IEEE Press, New York (2017). doi: 10.1109/CVPR.2017.243.
25. Otoja-Starzewski, M.: Material spatial randomness: From statistical to representative volume element. Probabilistic Engineering Mechanics. **21**, 112-132 (2006). doi: 10.1016/j.probengmech.2005.07.007

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77977-1_44

26. Kanit, T., Forest, S., Galliet, I., Mounoury, V., Jeulin, D.: Determination of the size of the representative volume element for random composites: statistical and numerical approach. International Journal of Solids and Structures. **40**, 3647 – 3679 (2003). doi: 10.1016/S0020-7683(03)00143-4
27. Mariani, S., Martini, R., Ghisi, A., Corigliano, A., Beghi, M.: Overall elastic properties of polysilicon films: a statistical investigation of the effects of polycrystal morphology. Journal for Multiscale Computational Engineering. **9**, 327-346 (2011). doi: 10.1615/IntJMultCompEng.v9.i3.50

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77977-1_44