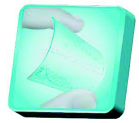
ASME Journal of Applied Mechanics
Online journal at:
https://asmedigitalcollection.asme.org/appliedmechanics



Convolutional Neural Network and Image Processing-Based Approach for Homogenization of Lattice Structures

Mohammed Abir Mahdi[1]
School of Mechanical and Aerospace Engineering,
Oklahoma State University,
Stillwater, OK 74078
e-mails: mmahdi@okstate.edu;
mmahdi@purdue.edu

Christopher Crick
Associate Professor
Department of Computer Science,
Oklahoma State University,
Stillwater, OK 74078
e-mail: chriscrick@cs.okstate.edu

Wei Zhao[2]
Associate Professor
School of Mechanical and Aerospace Engineering,
Oklahoma State University,
Stillwater, OK 74078
e-mail: wzhao@okstate.edu

The homogenization approach is widely used in lattice structure design to simplify modeling of complex geometries by representing them as simple solid elements in finite element analysis (FEA). Homogenized material properties are obtained through microscale analysis of a representative volume element (RVE). Several methods exist to compute effective properties, including beam theory, asymptotic homogenization, and the finite element method. FEA is commonly used for its ability to model arbitrary lattice geometries accurately. However, for complex structures, conventional FEA is computationally expensive, requiring extensive preprocessing and producing large stiffness matrices, which reduces efficiency in multiscale analysis and optimization. This study introduces an image-based machine learning framework for efficient microscale modeling of arbitrarily shaped lattice structures. Unlike traditional surrogate modeling approaches that rely solely on machine learning, the novelty of this method lies in maintaining consistency between physical modeling and machine learning representations. Specifically, the binary matrix obtained via image processing is used to generate the finite element model for asymptotic homogenization (AH)-based microscale analyses, whose outputs train a convolutional neural network (CNN). This eliminates the need for conventional mesh generation and ensures the network learns from physically consistent data. The CNN framework enables rapid and accurate prediction of effective material properties from image-based RVE representations. The developed CNN models predict the homogenized elastic properties with normalized mean absolute error below 2% and normalized mean squared error on the order of 10^{-4}, while requiring only a fraction of the computational time associated with conventional finite element-based homogenization analyses. [DOI: 10.1115/1.4071487]

Keywords: homogenization, machine learning, lattice structures, finite element analysis

1 Introduction

A lattice structure consists of spatially periodic elements such as rods, beams, plates, or shells, arranged systematically in tessellated patterns. These elements possess characteristic length scales typically larger than those associated with local load deformation. The fundamental motivation for developing lattice structures is to achieve designs that are lightweight, rigid, and efficient in terms of vibrational, acoustic, and thermal performance [1–3]. Owing to their exceptional strength-to-weight ratio and superior energy absorption capacity, lattice panels have gained increasing attention across multiple industries, including aerospace, automotive, biomedical, and marine engineering [4]. In aerospace and renewable energy systems, for instance, they are widely adopted in components such as aircraft wings and NASA rocket shell (as illustrated in Fig. 1) and wind turbine blades, where their low density and high impact resistance are critical. In the biomedical field, their porous architecture facilitates tissue ingrowth and allows for mechanical properties comparable to those of the surrounding bone, thereby minimizing stress shielding in orthopedic implants [7]. Beyond these applications, lattice cores have also revolutionized lightweight civil structures by serving as the core of sandwich panels. These lattice-core sandwich panels provide substantial weight reduction while maintaining high structural stiffness and strength, absorbing energy under impact loads, positioning them as a cutting-edge solution for modern engineering applications. Given these broad advantages, researchers continue to explore how different lattice geometries influence material properties relative to their fully solid counterparts. Variations in lattice topology can lead to significant differences in performance—some designs exhibit enhanced buckling resistance, while others demonstrate greater flexural rigidity [8]. Selecting the optimal lattice configuration thus remains a central challenge in the design process. With an extensive range of possible geometries spanning thousands

[1]Presently a PhD student in the School of Mechanical Engineering at Purdue University, West Lafayette, USA.
[2]Corresponding author.
Manuscript received May 12, 2025; final manuscript received March 9, 2026; published online April 6, 2026. Assoc. Editor: Xiaojia Shelly Zhang.

of known and yet-to-be-discovered configurations, lattice structures present a vast design space rich in research opportunities. As a result, this area of exploration has always been a longstanding focus of interest for the engineers and scientists working in the field of mechanical engineering and materials science.

Gibson and Ashby [9] were among the first to investigate the lattice structures for hexagonal honeycomb structures. In their study, they examined the behavior of lattice tube beam members formed by cell walls, examining aspects such as bending, elastic buckling, and plastic collapse. At the fundamental unit cell level, they employed beam theory-based homogenization to investigate the effective material properties of honeycomb lattice cores. Furthermore, they developed a precise mathematical solution for the mechanical characteristics of honeycomb, which was validated by extensive experimental testing. Subsequently, Masters and Evans [10] expanded the work studied by Gibson and Ashby [9] by integrating three mechanisms-flexure, stretching, and hinging, which enabled them to derive a more comprehensive expression for calculating effective material properties of honeycomb lattice cores. Wang and McDowell [11] investigated the lattice structures with several different cell types by using the same approach studied by Gibson and Ashby [9] and developed explicit mathematical equations for the effective in-plane mechanical properties. However, the analytical approach is limited by certain constraints [12], which is only suitable for small relative density ($r < 0.3$). The beam model is not accurate anymore when the lattice tube size increases with the relative density. Moreover, due to the assumption of small strains and the inability to account for large deformations, this beam theory-based explicit formulation cannot be applied to situations involving nonlinearity and complex geometric structures. Expanding mathematical solutions to broader scenarios has revealed several drawbacks in the context of beam theory-based homogenization. Consequently, numerical approaches, such as asymptotic homogenization (AH), have gained widespread acceptance in predicting mechanical properties of porous materials with arbitrary shaped lattice structures [13]. Guedes and Kikuchi [14] conducted a comprehensive study on this AH method-based material homogenization, and the application in lattice structure design via topology optimization [15,16]. Additionally, Arabnejad and Pasini [17] have conducted extensive research on employing AH method to determine the mechanical properties of lattice materials and structures and pointed out that AH has neither limitation on the cell topology nor on the range of relative density. The AH method can handle any lattice regardless of its relative density, which receives wide applications for lattice structures with any shape cell geometry.

However, it is important to note that the AH-based homogenization approach requires extensive computational time for both finite element mesh generation and analysis, particularly for lattice structures with complex shape periodic unit cells. Generally, there are two challenges associated with the homogenized constitutive properties calculation of complex lattice structure when using finite element method-based approaches: tedious mesh generation and fine mesh-based finite element analysis. Researchers employ sophisticated finite element analysis software such as ABAQUS [18] and ANSYS [19] to meticulously mesh or design these lattice structures and then conduct a fine-mesh-based microscale modeling analysis. Consequently, this intricate process leads to a substantial time investment during the lattice structure design, particularly FE^2 method-based multiscale modeling analysis [20]. For example, a lightweight wind turbine blade design may require different lattice structures across its blade length direction due to the nonuniform load distribution. Moreover, when researchers come across a novel lattice shape, they must carefully redesign the structure and repeat the same detailed process. Thus, the determination of the feasible lattice structure design for practical applications has become computationally expensive.

To address these challenges, there are some efforts that have been made towards using machine learning (ML)-based approaches for studying the homogenized constitutive properties of lattice structures. Settgast et al. [21] have used a hybrid multiscale neural network approach instead of traditional material modeling to calculate constitutive functions. The average volume method-based homogenization is employed to produce multiple datasets for training of a neural network. Alwattar and Mian [22] generated material constitutive properties data for body-centered cubic lattices via finite element simulation and used those data for training the neural network. However, the neural networks developed were specific to certain types of lattice shapes and applications. Gongora et al. [23] developed ML-based surrogate models to address the significant computational cost associated with the microscale analysis, whose ML-based surrogate model can iteratively and efficiently curate training datasets for optimization tasks. The finite element method is used to generate the training dataset. Ma et al. [24] used support vector regression-based ML for predicting the mechanical properties of 3D printed lattice structures. The porosity, material density, elastic modulus, and unit length of the lattice unit cell are combined with entropy as the inputs of the ML model. Experimental results under different parameters were used as the training dataset. Hooshmand et al. [25] used different ML algorithms to study the strain of lattice polymer foam under compression loads. The effective material properties of lattice structure-based polymer foam are studied

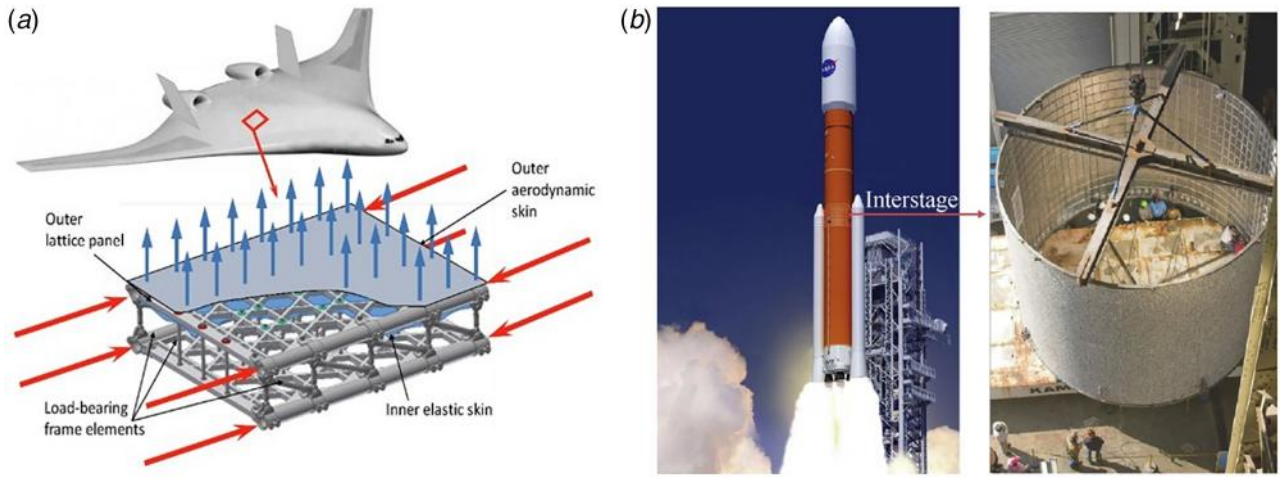


Fig. 1 Applications of lattice structures in Aerospace. (*a*) Lattice in airplane wing (reprinted from Dubovikov et al. [5], under the Creative Commons CC BY 4.0 license). (*b*) Lattice in NASA rocket shell (reprinted from Zhu et al. [6], under the Creative Commons CC BY-NC-ND 4.0 license).

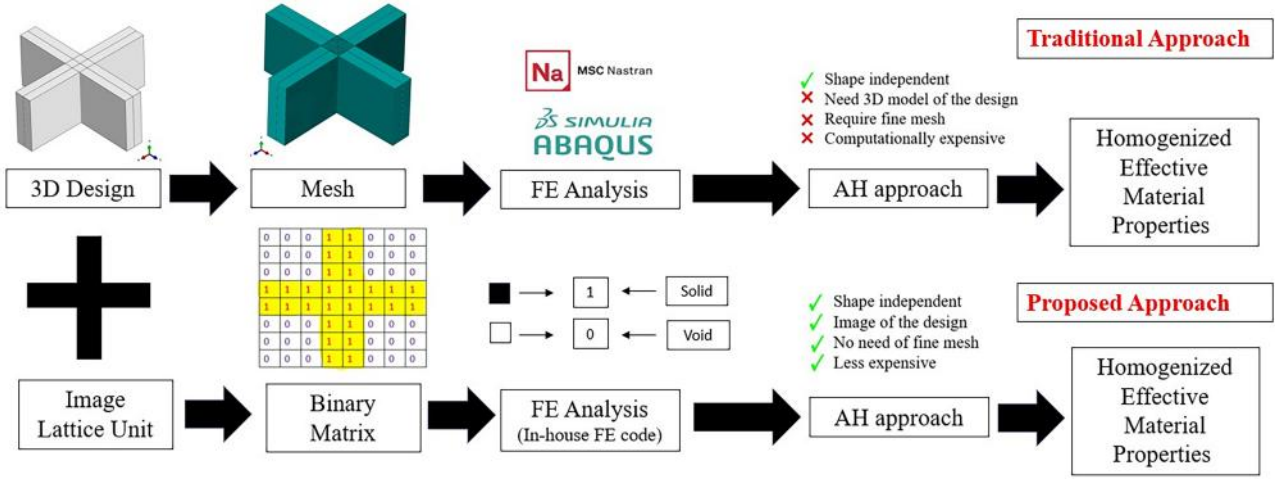


Fig. 2 Comparison between traditional and proposed approaches

under different parameters including lattice type, cell length across three axes, and the cell wall thickness. During the ML development process, the authors considered generating a dataset using finite element method to calculate the effective mechanical properties of lattice parts.

Some research studies have been conducted to use ML techniques to find out the optimal distribution of the lattice unit cell for a desired material design. Wang and Panesar [26] developed a neural network (NN)-based inverse lattice generator, which is trained to output lattice unit cells from the input of target mechanical properties. The proposed ML-based lattice generation method utilizes the density distribution and stress field of low-resolution topology optimization (TO) design to inform the inverse generator and produce lattice cells. A few geometric parameters were defined to parameterize the shape of the lattice structures, which was used during the ML training process. Yu et al. [27] used an artificial neural network (ANN) to study the mechanical behaviors of heterogeneous lattice structures. They proposed a binarization method to facilitate the digital representation of each lattice configuration where one represents the reinforcement phase and zero represents the matrix phase. The flattened binary matrix was used in the input layers during the ANN training process. The well-trained artificial neural network is able to generate heterogeneous lattice structures to meet various specific target performances. Ibrahimi et al. [28] developed ML models to study the relation between input parameters (analytical) and the mechanical and morphological properties of the triply periodic minimal surfaces (TPMS) structures. A dataset with more than one thousand TPMS scaffolds was considered for the training by using the data obtained from finite element modeling (FEM). The trained model is able to identify the microstructure of TPMS scaffolds, which simultaneously satisfies multidisciplinary requirements.

Inspired by topology optimization [29], lattice structure can be designed from a solid structure using binary variables (density method) whose shape does not depend on any prescribed parameterization method. The binary mapping can be obtained by using image processing, along with the recent development of using ML for image segmentation [30], both provide an opportunity to use an image to characterize the lattice structure rather than using physical parameters during ML development. The present study introduces an innovative framework for rapidly determining the homogenized constitutive properties of lattice structures. Unlike existing ML development for studying lattice structures' stiffness properties, the novelty of this work lies in maintaining complete consistency between the physical modeling and the ML representations. Specifically, the binary matrix obtained through image processing is directly used to generate the FE model for conducting AH-based microscale analyses, whose results are then used to train a convolutional neural network (CNN). In this framework, the CNN interprets the image of the lattice structure as a matrix of pixel values, providing a binary mapping identical to that used in the FE model. Importantly, the mesh is developed not only for the solid lattice structure but for the entire domain including the void regions, thereby ensuring that the physical model and the ML model share an identical geometric representation. By unifying the physical and computational representations, the proposed approach establishes a novel paradigm for developing a generalized neural network capable of accurately and efficiently predicting the effective material properties of lattice structures. To demonstrate this methodology, the AH-based finite element analysis is applied to 2D lattice structures. The paper is organized as follows: Section 2 details the methodology for generating finite element models from image processing and validating the AH-based homogenization. Section 3 describes the development of the CNN model. Section 4 presents the results and discussion, and Sec. 5 summarizes the key findings and outlines future research directions.

2 Methodology

2.1 Image Processing-Based Homogenization Analysis. This section discusses the use of image processing technique to generate finite element models for conducting material homogenization analysis of lattice structures. To demonstrate the proposed approach, a two-dimensional image of the lattice structure representative volume element (RVE) is employed. If the RVE exhibits a uniform cross section through its thickness, constructing a detailed three-dimensional model is unnecessary to capture geometric features such as member thickness. In such cases, a two-dimensional image adequately represents the model of the RVE for finite element analysis. As illustrated in Fig. 2, the traditional approach generates the mesh for the lattice structures and then employs finite element analysis software to calculate the effective material properties. The proposed approach uses the domain (design domain used in the topology optimization) to develop the finite element model. This modeling process uses the image pixel feature, which will be consistent with those used by CNN in ML, which keeps the consistency of the models used in the physical modeling and ML training. Like the topology optimization of lattice structures, a regular model consisting of shell elements is used to represent the whole domain. The value of relative density, "1," represents the presence of material where "0" represents void. Such binary mapping enables this study to intentionally employ a black-and-white image of RVE, facilitating binary conversion where solid pixels (material) are assigned a value of "1"

and void pixels are assigned "0" for subsequent element properties definition. The finite element mesh size is the same as the pixel resolution size, enabling the use of the binary mapping for finite element model development. Since the AH approach is inherently shape independent, it can handle complex geometries; however, it traditionally requires a fine mesh when the shape of lattice shape of RVE is complex. This methodology further enables the construction of a comprehensive dataset of homogenized constitutive properties for various lattice configurations.

2.2 Homogenization. Homogenization involves identifying the mechanical properties of a heterogeneous lattice medium by using an RVE. It possesses the microstructural properties of effective materials and can be used to determine the structural performance of the whole domain. The similar processes can be applied to determine the effective material properties of any other shapes of unit cells. The homogenization process applied to a lattice with square unit cells is schematically shown in Fig. 3. It can be illustrated from the figure that a homogenized domain $\bar{\Omega}$ replaces a domain Ω with a periodic microstructure that has been exposed to traction t at the traction boundary Γ_t, displacement d at the displacement boundary Γ_d, and body force f. Both the discrete and homogenized medium, the latter being a homogeneous continuum devoid of voids, which is subject to the external and traction bounds. The macroscopic behavior of Ω and $\bar{\Omega}$ should behave similarly when determining the effective mechanical characteristics of the RVE. In addition to square unit cells, similar processes can be applied to determine the effective material properties of other shapes of unit cells.

2.3 Asymptotic Homogenization. AH provides a rigorous framework to link micro- and macro-scales. In AH, every field quantity depends on two independent scales: a macroscopic scale $\boldsymbol{x}$, which varies slowly over the domain, and a microscopic scale $\boldsymbol{y} = \boldsymbol{x}/\varepsilon$, which captures rapid variations within the RVE. The small parameter ε is the ratio of the characteristic length of the RVE to the macroscopic domain. It is not a direct scaling of fields; rather, it allows the governing equations to be expanded asymptotically in powers of ε, with each term capturing progressively finer details. Because $\varepsilon \ll 1$, derivatives with respect to $\boldsymbol{y}$ are amplified by a factor of $1/\varepsilon$, reflecting the fact that stress and strain fluctuate rapidly at the microscale, while the macroscopic response varies smoothly ($d/d\boldsymbol{x} = \partial/\partial\boldsymbol{x} + (1/\varepsilon)(\partial/\partial\boldsymbol{y})$). The macroscopic fields (e.g., displacement) are smooth functions of x, capturing the averaged behavior over many unit cells, while the microscopic fields are varying periodically for each RVE. Essentially, the macroscopic solution provides the envelope, and the microscopic solution describes the repeating detailed fluctuations inside the material. This dual-scale decomposition allows accurate prediction of the effective mechanical response, independent of the unit cell shape or relative density, making AH more general and reliable than beam-theory-based homogenization methods [12]. Moreover, the AH is one of the most reliable and accurate

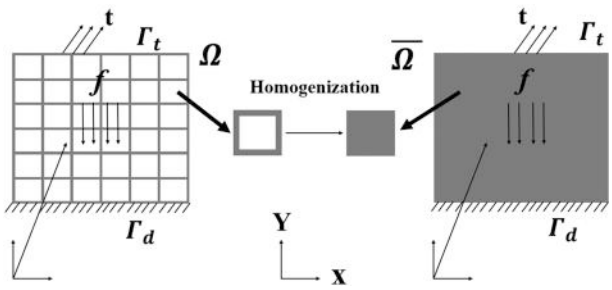


Fig. 3 An illustration of a square lattice structure (left) into a homogenized medium (right) with effective properties determined by the RVE [31]

approaches verified using experimental data [32]. The macroscopic elasticity tensor E^H_{ijkl} of a periodic lattice is obtained as a volume average over the RVE [33]. This formulation is based on the principle of minimum potential energy, whereby the homogenized macroscopic elastic energy is computed as the averaged strain energy of the heterogeneous microstructure. Thus,

$$E^H_{ijkl} = \frac{1}{|V|}\int_V E_{pqrs}\left(\varepsilon^{0(ij)}_{pq} - \varepsilon^{(ij)}_{pq}\right)\left(\varepsilon^{0(kl)}_{rs} - \varepsilon^{(kl)}_{rs}\right)\mathrm{d}V \tag{1}$$

where $|V|$ denotes the volume of the unit cell, E_{pqrs} is the locally varying stiffness tensor, $\varepsilon^{0(ij)}_{pq}$ are prescribed macroscopic strain fields (in 2D, there are three; e.g., unit strain in the horizontal direction (11), unit strain in the vertical direction (22), and unit shear strain (12 or 21)), and the locally varying strain fields $\varepsilon^{(ij)}_{pq}$ are defined as

$$\varepsilon^{(ij)}_{pq} = \varepsilon_{pq}\left(\chi^{ij}\right) = \frac{1}{2}\left(\chi^{ij}_{p,q} + \chi^{ij}_{q,p}\right) \tag{2}$$

The displacement fields χ^{kl} are determined by solving the elasticity equations at the microscale with applied macroscopic strains:

$$\int_V E_{ijpq}\varepsilon_{ij}(v)\varepsilon_{pq}(\chi^{kl})\,dV = \int_V E_{ijpq}\varepsilon_{ij}(v)\varepsilon^{0(kl)}_{pq}\,dV \quad \forall v \in V \tag{3}$$

where v is a virtual displacement field. This equation is typically solved numerically using the finite element method. For a 2D RVE, the effective elastic stiffness matrix can be expressed in terms of the homogenized Young's moduli, Poisson's ratios, and shear modulus:

$$E^H = \begin{bmatrix} \dfrac{E^{\text{eff}}_{xx}}{1-\nu^{\text{eff}}_{xy}\nu^{\text{eff}}_{yx}} & \dfrac{\nu^{\text{eff}}_{xy}E^{\text{eff}}_{yy}}{1-\nu^{\text{eff}}_{xy}\nu^{\text{eff}}_{yx}} & 0 \\ \dfrac{\nu^{\text{eff}}_{xy}E^{\text{eff}}_{yy}}{1-\nu^{\text{eff}}_{xy}\nu^{\text{eff}}_{yx}} & \dfrac{E^{\text{eff}}_{yy}}{1-\nu^{\text{eff}}_{xy}\nu^{\text{eff}}_{yx}} & 0 \\ 0 & 0 & G^{\text{eff}}_{xy} \end{bmatrix} \tag{4}$$

where E^{eff}_{xx} and E^{eff}_{yy} are the homogenized Young's moduli along x and y directions, ν^{eff}_{xy} and ν^{eff}_{yx} are the homogenized Poisson's ratios, and G^{eff}_{xy} is the homogenized shear modulus.

2.4 Validation of Image Processing-Based Model. A validation study is conducted to assess whether the binary mapping of the images can accurately represent the RVE model in the AH homogenization study. The mesh size is the same as the image pixel size, a high resolution image is used for converged results. In this study, 100×100 pixels is considered for the domain to ensure mesh convergence of the effective material properties. To demonstrate this binary mapping-based finite element model development, examples with varying wall thicknesses of the RVEs are employed. The comparison between the present results and literature values [34] is presented in Table 1. It is clearly seen that the present results using a binary mapping-based finite element mesh for AH homogenization analysis of lattice structures are close to the literature results [34], where the traditional finite element method and energy-based homogenization approach were used. The components of the effective elastic tensor are close to each other with slight differences.

The slight difference can be attributed to the small difference in the finite element model generated using the traditional finite element model and that obtained using binary mapping of the lattice RVEs. A finite element mesh representation of the binary matrix of the first RVE from Table 1 is shown in Fig. 4(*a*). The denoted white circles illustrate the missing of elements due to the binary mapping at the sharping corners of the lattice structure, which leads to the emergence of minor differences in components of the effective elastic tensor matrix. A finer and more accurate image would likely eliminate these discrepancies. A parametric image size convergence study further confirmed that refining the

Table 1 Comparison of effective elastic tensor between the literature [34] and the present study for different square unit RVEs

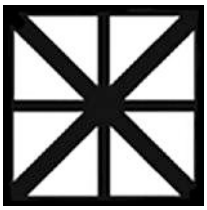
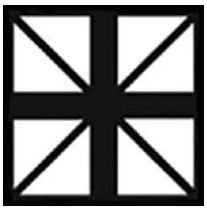
RVE shape	Study	Effective elastic tensor, E^H (MPa)	Average error (%)
	Literature	$\begin{bmatrix} 236.29 & 107.57 & 0.00 \\ 107.57 & 236.29 & 0.00 \\ 0.00 & 0.00 & 101.99 \end{bmatrix}$	–
	AH	$\begin{bmatrix} 237.67 & 108.30 & 0.00 \\ 108.30 & 236.23 & 0.00 \\ 0.00 & 0.00 & 100.54 \end{bmatrix}$	0.376
	Literature	$\begin{bmatrix} 260.05 & 83.19 & 0.00 \\ 83.19 & 260.05 & 0.00 \\ 0.00 & 0.00 & 78.09 \end{bmatrix}$	–
	AH	$\begin{bmatrix} 261.24 & 83.60 & 0.00 \\ 83.60 & 259.43 & 0.00 \\ 0.00 & 0.00 & 75.76 \end{bmatrix}$	0.518
	Literature	$\begin{bmatrix} 274.24 & 70.06 & 0.00 \\ 70.06 & 274.24 & 0.00 \\ 0.00 & 0.00 & 64.50 \end{bmatrix}$	–
	AH	$\begin{bmatrix} 276.95 & 71.56 & 0.00 \\ 71.56 & 275.33 & 0.00 \\ 0.00 & 0.00 & 63.11 \end{bmatrix}$	0.869
	Literature	$\begin{bmatrix} 290.08 & 56.81 & 0.00 \\ 56.81 & 290.08 & 0.00 \\ 0.00 & 0.00 & 48.71 \end{bmatrix}$	–
	AH	$\begin{bmatrix} 297.65 & 59.30 & 0.00 \\ 59.30 & 293.57 & 0.00 \\ 0.00 & 0.00 & 48.97 \end{bmatrix}$	1.457

image resolution can significantly reduce this difference. However, due to the high computational cost and the already satisfactory accuracy in determining the effective elastic tensor, the present study continued using images with a resolution of 100×100 pixels. An image representation of the finite element symmetric mesh of the cross-shaped RVE is shown in Fig. 4(*b*), which can accurately capture the shape of lattice structures, while binary mapping has been shown to be highly effective for stiffness-dominated problems where global compliance is relatively insensitive to fine geometric details, which may pose some challenges for strength-dominated analyses. The stair-step boundaries and sharp edges inherent to the pixelation process can introduce artificial stress concentrations, thereby diminishing the accuracy of local stress predictions essential for evaluating material failure and structural integrity. Therefore, for strength-critical applications, future work will consider refining the binary-mapped geometry through surface smoothing.

2.5 Dataset Generation. In ML applications, success depends on having a good curated dataset that serves as the basis for model development [35]. A high-quality dataset not only provides essential raw information for accurate predictions and meaningful pattern recognition, but it also ensures the generalization of ML algorithms to handle novel data effectively. By identifying pertinent features and addressing biases, high-quality dataset's diverse and precise nature enhances model efficiency and promotes fairness. After successfully verifying the developed code for the AH approach in Sec. 2.4, a comprehensive dataset was generated, which comprised material properties for 10,000 lattice images.

To initiate this process, a square lattice RVE (1×1 unit) with walls (0.20 unit thickness) along the diagonals was selected. The specific material properties used for the lattice RVE are presented in Table 2. A black wall with a thickness of 1 pixel surrounds all of the lattice images. Afterward, the selected RVE was augmented-rotated at various angles between 0 and 360 deg, resulting in the creation of 10,000 unique images. A 100×100 computational grid remained fixed while the material geometry was rotated rigidly over this grid. As shown in Table 3 and Fig. 5, the rotational augmentation leads to noticeable variations in the discretized image designs. These variations are manifested as small changes in the black (material) and white (void) area

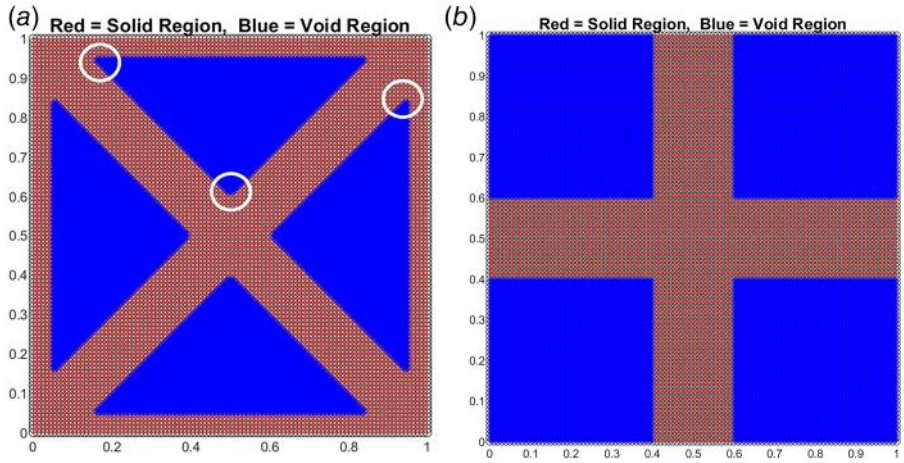


Fig. 4 FEM mesh of lattice structures obtained from binary mapping. (*a*) FEM mesh from binary matrix (with missing elements). (*b*) FEM mesh from binary matrix (no missing elements).

fractions, arising from the fixed grid discretization. The image rotation process has been randomized within the specified range of angles so that the images do not remain equidistant, allowing for unique orientations and variations in each generated image. Although, due to double symmetry, it would be sufficient to consider only angles between 0 and 90 deg, the study wanted to develop a generalized algorithm capable of performing the analysis even for nonsymmetric lattice designs. Although any size of the dataset may appear to have a risk of overfitting, this issue is addressed in the CNN model through the inclusion of dropout layers during training. It is a regularization technique used in CNNs and other neural networks to prevent overfitting. These images were then processed through the FEM code employing the AH approach, enabling the extraction of material properties data for each image. A flowchart is shown to demonstrate this process, as seen in Fig. 6. The large-scale database generation was conducted using the computational resources available in the Oklahoma State University High Performance Computer Center.

3 Convolutional Neural Network

CNNs are a class of deep learning architectures inspired by the structure and function of the visual cortex in biological systems [36]. Originally developed for image recognition tasks, CNNs have since found widespread application in fields such as object tracking, text recognition, medical imaging, and material microstructure analysis [37–40]. Conceptually, CNNs extend the structure of traditional artificial neural networks by introducing spatially localized connectivity: each neuron is connected only to a small subset of neurons in the preceding layer [41]. This sparse connectivity allows the network to efficiently learn spatial hierarchies of features including edges, shapes, and patterns, which are particularly valuable for identifying and classifying complex image-based data such as lattice geometries.

In this study, a CNN framework is developed to establish a data-driven mapping between lattice geometries and their corresponding homogenized material properties. The dataset consists of two-dimensional lattice images and their numerically derived mechanical properties. To ensure generalization, 80% of the dataset is used for model training, 10% for validation, and the remaining 10% for testing the network performance. Each image is of size 100×100 pixels, defining the input layer dimensions as $100 \times 100 \times 1$. Although this study focuses on two-dimensional lattices, the same framework can be extended to analyze three-dimensional structures in future work. Figure 7 shows the architecture that begins with a sequence of alternating convolutional and pooling layers that extract and progressively compress hierarchical geometric features. The notation m, n, N, and 5 denotes the image height ($m = 100$ pixels), image width ($n = 100$ pixels), the number of augmented images ($N = 10{,}000$ for the present study), and the five target material properties, respectively.

The network's first stage uses a convolutional layer with 64 filters of size 3×3. Convolutional layers serve two primary purposes: (1) they learn local patterns such as edges or short struts in lattice images by applying learned kernels across the spatial domain and (2) they reuse the same kernel parameters across the image, dramatically reducing the number of free parameters compared with fully connected layers. The 3×3 kernel is a standard choice that balances locality (capturing small-scale geometric features) and computational cost. The study selected 64 filters in the first layer to allow the network to represent a moderate variety of low-level features while keeping the parameter count manageable. Each convolutional stage is followed by a 2×2 max-pooling operation. Pooling reduces the spatial resolution of feature maps, which (a) increases the effective receptive field of later layers so they can capture larger-scale patterns, (b) reduces computational cost and memory usage, and (c) provides a small degree of translational invariance, which is useful when the exact pixel location of a feature is not critical for predicting homogenized properties. Max-pooling is used here because it preserves the strongest activations (dominant local features), which are often the most informative for geometry recognition. The convolution and pooling pattern is repeated for three additional convolutional layers with increasing filter counts (128, 256, and 512). The rationale for progressively increasing filter numbers is twofold: earlier layers learn simple,

Table 2 Material properties [33]

Young's modulus, E_s (GPa)	1
Poisson's ratio, ν_s	0.30

Table 3 Black (material) and white (void) area fractions for the representative dataset images

Image no.	Black pixels	White pixels	Black ratio	White ratio
1	5154	4846	0.5154	0.4846
5000	5286	4714	0.5286	0.4714
9006	5140	4860	0.5140	0.4860

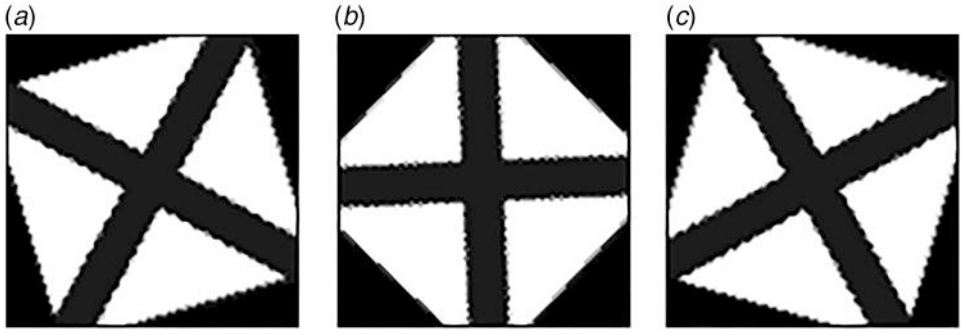


Fig. 5 Representative images from the developed dataset. (*a*) First image from the developed dataset. (*b*) 5000th image from the developed dataset. (c) 9006th image from the developed dataset.

low-level features (edges, corners), while deeper layers must represent more abstract and diverse patterns (assemblies of struts, cell topologies). Increasing the number of filters with depth gives the model greater representational capacity where it is most needed. The exact filter counts and number of layers were chosen through iterative tuning using the validation set: the study increased capacity until validation performance improvements saturated or overfitting trends appeared, then selected the smallest architecture that met the accuracy/generalization trade-off.

After the final convolutional block, the multidimensional feature maps are flattened into a one-dimensional vector and passed to a sequence of fully connected (dense) layers. Dense layers perform high-level nonlinear combinations of the features extracted by the convolutional backbone and are therefore appropriate where the task requires mapping a compact global representation to the final continuous outputs (material properties). The first dense layer contains 512 neurons: this relatively large layer projects the flattened convolutional output into a high-capacity representation capable of modeling complex relationships between geometry-derived features and mechanical responses. The network then reduces dimensionality through subsequent dense layers with 256 and 128 neurons, respectively. This progressive reduction ($512 \rightarrow 256 \rightarrow 128$) acts as a learned funnel that concentrates information and helps the model form increasingly compact representations before the final regression. Reducing the number of neurons in successive layers also helps control overfitting by limiting parameter growth while preserving sufficient capacity to model the mapping. As with the convolutional filters, the specific neuron counts were determined empirically by cross-validation and by monitoring the number of trainable parameters relative to dataset size (with $N = 10{,}000$ augmented images, the chosen sizes provided a good balance of capacity and generalization).

To reduce overfitting, each dense layer (except the output) is followed by a dropout layer with dropout rate 0.5. Dropout randomly sets a fraction of activations to zero during training, forcing the network to avoid reliance on any single neuron and thereby improving robustness. The relatively high dropout rate was found to be effective in regularizing the model, given the problem complexity and available training samples.

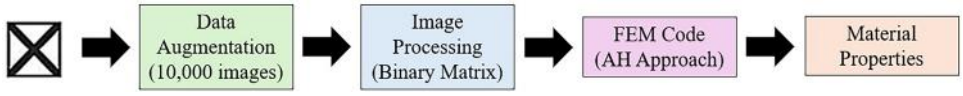


Fig. 6 An illustration of the process used to determine the material properties of different lattice unit cells, including a visual representation of the FEM mesh

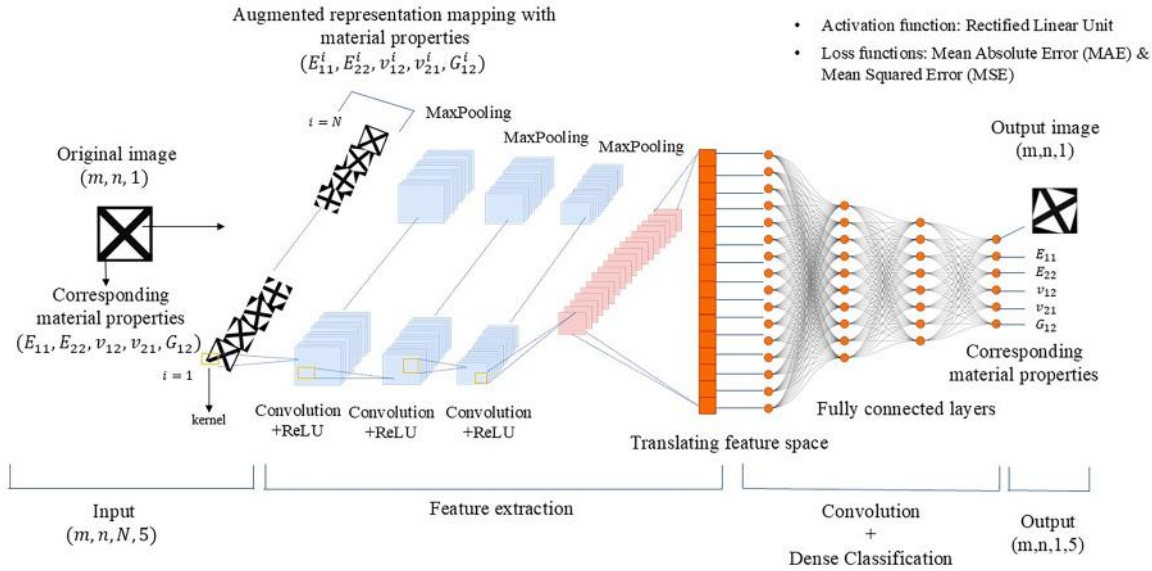


Fig. 7 Generalized architecture of the developed CNN model

All hidden layers use the Rectified Linear Unit (ReLU) activation function. ReLU is defined as $ReLU(x) = \max(0, x)$, and it is used here for several important reasons: (1) Computational efficiency: ReLU is cheap to evaluate and differentiable almost everywhere, which accelerates both forward and backward passes during training. (2) Mitigation of the vanishing gradient problem: unlike saturating activations (e.g., sigmoid or tanh), ReLU preserves gradients for positive inputs, allowing effective training of deep networks. (3) Sparse activations: ReLU outputs exact zeros for negative inputs, which induces sparsity in activations; this sparsity can improve representational efficiency and acts as an implicit regularizer. ReLU was therefore chosen for its blend of practical advantages and suitability for deep convolutional architectures in image-based regression tasks [42]. The final output layer uses a linear activation because the network performs multi-output regression: it must predict five continuous material properties. A linear output layer allows the network to produce unbounded, real-valued predictions appropriate for regression targets. The model is trained using a loss function suited to continuous outputs (e.g., mean-squared error) and standard optimization algorithms implemented with *TensorFlow/Keras*. Implementation details are presented in the following. The model was built using *TensorFlow and Keras*, and preprocessing/visualization pipelines use *NumPy, Pandas, OpenCV, Scikit-learn, and Matplotlib*. Hyperparameters (filter counts, layer depths, neuron counts, dropout rates, learning rate, and batch size) were selected through iterative tuning on the validation set with attention to model complexity relative to the available augmented data (10,000 images).

3.1 Hyperparameter Selection

3.1.1 Epoch Size Selection. Selecting an appropriate number of training epochs is essential to ensure efficient convergence without overfitting [43]. For every tested combination of batch size (16, 32, 64, and 128), learning rate (0.01, 0.001, and 0.0001), and optimizer (*Adam, SGD, RMSprop, Adagrad, Adadelta, and Nadam*), the network was trained for up to 80 epochs (presented in Appendices A and B) while monitoring its performance on a separate validation set. At the end of each epoch, the validation error—validation mean absolute error (MAE) for MAE loss training and validation mean squared error (MSE) for MSE loss training—was recorded. The optimal number of epochs was defined as the epoch at which this validation error reached its minimum value, corresponding to the point of best generalization before overfitting occurred. Using this criterion, the MAE loss model achieved its lowest validation MAE at epoch 29, while the MSE loss model reached its minimum validation MSE at epoch 39, and these values were selected as the optimal training durations for the respective runs.

3.1.2 Learning Rate Selection. The learning rate was selected by comparing candidate values (0.01, 0.001, and 0.0001) based solely on validation performance across all batch sizes and optimizers. For each learning rate, the smallest validation MAE and validation MSE achieved by any configuration using that learning rate were identified. Because MAE and MSE have different numerical scales, both metrics were normalized to a common 0–1 range using min–max normalization, and the normalized values were summed to form a combined normalized score. The learning rate with the lowest combined score was selected, as it simultaneously minimized both validation error measures in a balanced manner. Following this procedure, a learning rate of 0.001 was identified as optimal for both the MAE loss and MSE loss runs.

3.1.3 Optimizer Selection. After fixing the learning rate at 0.001, the optimizer was selected by directly comparing validation performance under identical conditions. The analysis focused primarily on the *Adam* and *Nadam* optimizers, which consistently exhibited the most stable convergence during the sweep. For each candidate optimizer, the best validation MAE and MSE were extracted, normalized, and combined into a single validation-based score. The optimizer yielding the lower combined score was selected, indicating superior overall validation accuracy. Using this criterion, *Nadam* outperformed the alternatives and was, therefore, selected as the final optimizer for both the MAE-loss and MSE-loss models.

3.1.4 Batch Size Selection. The batch size was selected at the chosen learning rate and optimizer by evaluating validation performance across the tested batch sizes (16, 32, 64, and 128). For each batch size, the best validation MAE and MSE achieved with the selected optimizer were identified, normalized, and combined into a single validation-based score. The batch size with the lowest combined score was chosen, as it provided the most favorable balance between training stability and generalization. This procedure resulted in a batch size of 32 for the MAE loss model and 16 for the MSE loss model, reflecting differences in how the two loss functions respond to gradient noise and mini-batch averaging. Comparison of the training and validation error convergence for the *Nadam* optimizer using the MAE and MSE loss functions is shown in Figs. 8(*a*) and 8(*b*), respectively. The validation curves fall below the training curves because dropout is active during training but disabled during validation, making the validation pass inherently easier. In addition, validation metrics are computed on the full dataset without mini-batch noise, producing smoother and slightly lower errors than the training metrics.

After identifying the optimal hyperparameters through validation-based selection, the first step was to construct the definitive CNN models and obtain an unbiased estimate of their predictive capability. To achieve this, the training and validation datasets were combined to maximize the amount of reliable information available for learning, and two final models were retrained from scratch: one using the MAE loss and the other using the MSE loss. Each model was trained using its respective optimal optimizer, learning rate, batch size, and number of epochs determined during the hyperparameter study, with training intentionally stopped at the optimal epoch to prevent overfitting. The resulting models were then evaluated once on a held-out 10% test subset from the original dataset, which had not been used during training or model selection. This step provides a single, unbiased benchmark of overall predictive accuracy and serves to confirm that the selected training configuration generalizes beyond the data used for optimization.

Comparing the two training strategies, both the MAE loss and MSE loss models demonstrate strong predictive performance on 10% unseen data, with average errors well below 2% in normalized space. The MAE loss model achieves a normalized MAE of approximately 1.74% and an MSE of about 0.049%, reflecting robust and stable predictions with uniform weighting of errors. The MSE loss model, on the other hand, attains a lower normalized MAE of approximately 1.59% and a reduced MSE of about 0.041%, indicating improved accuracy and more effective suppression of larger deviations.

While this initial test evaluation provides a global performance measure, it does not fully characterize how the models behave across individual material properties. Therefore, a second, complementary evaluation step was carried out using a completely separate dataset of 1000 previously unseen RVE images. In this stage, the two saved final models were loaded and applied without any further training. Because the reference properties in this dataset were already provided in normalized form, predictions were assessed directly in normalized space. Detailed error metrics —including MAE, MSE, square root of the mean squared error, and the coefficient of determination (R^2)—were computed for each property and plots were generated to visually compare predicted and numerical values obtained by the AH-based homogenization approach in Sec. 4.

3.1.5 Computational Cost. Table 4 shows the comparison of computational time used by AH homogenization including mesh

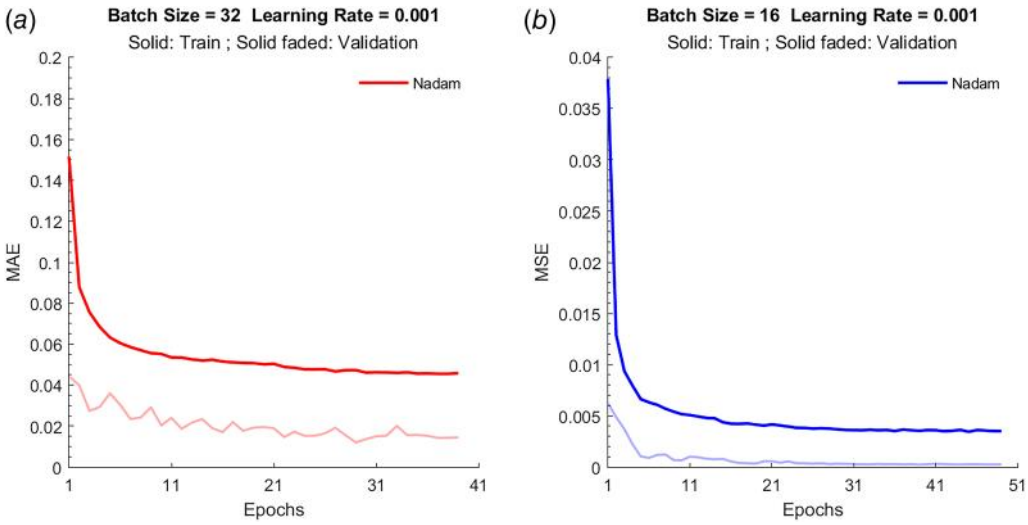


Fig. 8 Comparison of training and validation error convergence for *Nadam* optimizer using (a) MAE loss and (b) MSE loss. (*a*) Training and validation of MAE loss function for Nadam optimizer under the selected learning rate and batch size. (*b*) Training and validation of MSE loss function for *Nadam* optimizer under the selected learning rate and batch size.

generation and the ML evaluations. It shows that the AH approach takes about 26 seconds to calculate the material properties of a single lattice structure. In contrast, the developed ML model can determine the material properties of 1000 unit RVEs within just approximately 24 s, each around 0.024 s. This capability enables us to quickly navigate through the material properties of millions of lattice designs within short time, enabling us to identify those designs that are most appropriate for a particular application.

4 Results and Discussion

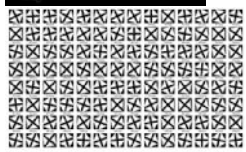
The normalized effective material properties predicted by the developed CNN model are presented in Figs. 9–13. In these figures, the normalized reference values obtained using the AH approach are compared with the corresponding normalized CNN predictions. The dashed 45-deg line represents perfect agreement between the actual and predicted normalized effective material properties.

Figure 9 presents the normalized effective longitudinal Young's modulus, E_{xx}^{eff}. The predictions obtained from both CNN models show excellent agreement with the AH-based normalized reference values, as evidenced by high coefficients of determination and low prediction errors. For the MAE-trained model, the normalized coefficient of determination reaches $R^2 = 0.9945$, indicating a strong linear correlation with the reference solution. The normalized mean absolute error is 1.81%, while the corresponding normalized mean squared error is 5.24×10^{-4}, resulting in a root mean squared error of approximately 2.29%. For the MSE-trained model, a comparable level of agreement is observed, with a normalized coefficient of determination of $R^2 = 0.9939$. The normalized mean absolute error is slightly higher at 1.86%, while the normalized mean squared error is 5.82×10^{-4}, corresponding to a root mean squared error of approximately 2.41%. Both models achieve prediction errors on the order of 1–2%.

A similar trend is observed for the normalized effective transverse Young's modulus, E_{yy}^{eff}, as shown in Fig. 10. For the MAE-trained model, the normalized coefficient of determination reaches $R^2 = 0.9945$, indicating a strong linear correlation with the reference solution. The normalized mean absolute error is 1.79%, while the corresponding normalized mean squared error is 5.13×10^{-4}, resulting in a root mean squared error of approximately 2.27%. For the MSE-trained model, a comparable level of agreement is observed, with a normalized coefficient of determination of $R^2 = 0.9941$. The normalized mean absolute error is slightly higher at 1.83%, while the normalized mean squared error is 5.60×10^{-4}, corresponding to a root mean squared error of approximately 2.37%. Both models achieve prediction errors on the order of 1–2%. While the MSE-trained model maintains comparable predictive fidelity, the MAE-trained model exhibits slightly lower average error measures, indicating that both loss formulations provide reliable and complementary performance for homogenized property prediction.

Figure 11 illustrates the comparison for the normalized in-plane Poisson's ratio ν_{xy}^{eff}, where the predicted values from both loss formulations exhibit a strong alignment with the ideal 45-deg reference line. For the MAE-trained model, the normalized coefficient of determination reaches $R^2 = 0.9942$, indicating a strong linear correlation with the reference solution. The normalized mean absolute error is 1.74%, while the corresponding normalized mean squared error is 4.42×10^{-4}, resulting in a root mean squared error of approximately 2.10%. For the MSE-trained model, a comparable level of agreement is observed, with a normalized coefficient of determination of $R^2 = 0.9951$. The normalized mean absolute error is slightly lower at 1.52%, while the normalized mean squared error is 3.78×10^{-4}, corresponding to a root mean squared error of approximately 1.94%. Both models achieve prediction errors on the order of 1–2%.

Table 4 Comparison of computational time using image of the unit RVEs

RVE shape	Study	Time (s)
	AH	26.05
 (1000 testing RVEs)	ML	23.93

A similar trend is observed for the transverse Poisson's ratio ν_{yx}^{eff}, as shown in Fig. 12. For the MAE-trained model, the normalized coefficient of determination reaches $R^2 = 0.9943$, indicating a strong linear correlation with the reference solution. The normalized mean absolute error is 1.73%, while the corresponding normalized mean squared error is 4.39×10^{-4}, resulting in a root mean squared error of approximately 2.10%. For the MSE-trained model, a comparable level of agreement is observed, with a normalized coefficient of determination of $R^2 = 0.9951$. The normalized mean absolute error is slightly lower at 1.52%, while the normalized mean squared error is 3.79×10^{-4}, corresponding to a root mean squared error of approximately 1.95%. Both models achieve prediction errors on the order of 1–2%.

The plot for the normalized effective shear modulus G_{xy}^{eff} is shown in Fig. 13, where the CNN-predicted values are compared against the corresponding results obtained from the AH approach. The predictions from both MAE- and MSE-trained models closely follow the ideal 45 deg reference line across the entire normalized range, indicating strong agreement between the numerical and machine-learning-based estimates. For the MAE-trained model, the normalized coefficient of determination reaches $R^2 = 0.9966$, indicating a strong linear correlation with the reference solution. The normalized mean absolute error is 1.41%, while the corresponding normalized mean squared error is 3.22×10^{-4}, resulting in a root mean squared error of approximately 1.79%. For the MSE-trained model, a comparable level of agreement is observed, with a normalized coefficient of determination of $R^2 = 0.9966$. The normalized mean absolute error is slightly lower at 1.49%, while the normalized mean squared error is 3.21×10^{-4}, corresponding to a root mean squared error of approximately 1.79%. Both models achieve prediction errors on the order of 1–2%. The results demonstrate that the developed CNN model is capable of accurately capturing the effective shear response of lattice unit RVEs from image-based inputs. Finally, among all images, the image for the best lattice structure RVE with the best material properties (e.g., highest Young's modulus) is shown in Fig. 14.

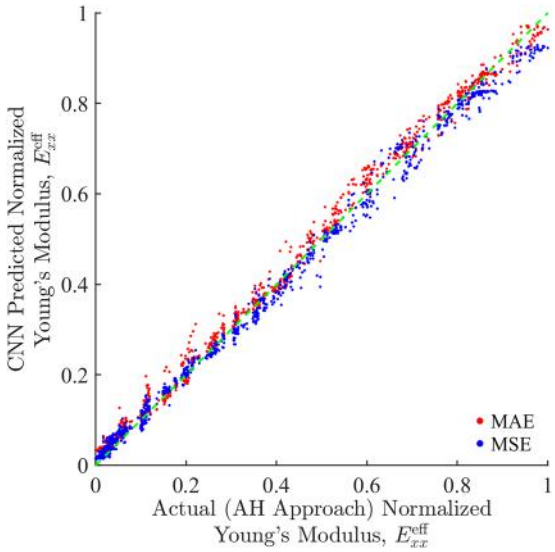


Fig. 9 Comparison between actual and predicted normalized Young's modulus E_{xx}^{eff}

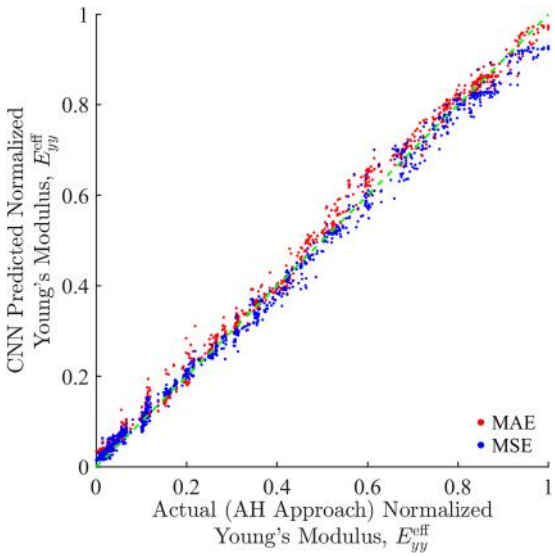


Fig. 10 Comparison between actual and predicted normalized Young's modulus E_{yy}^{eff}

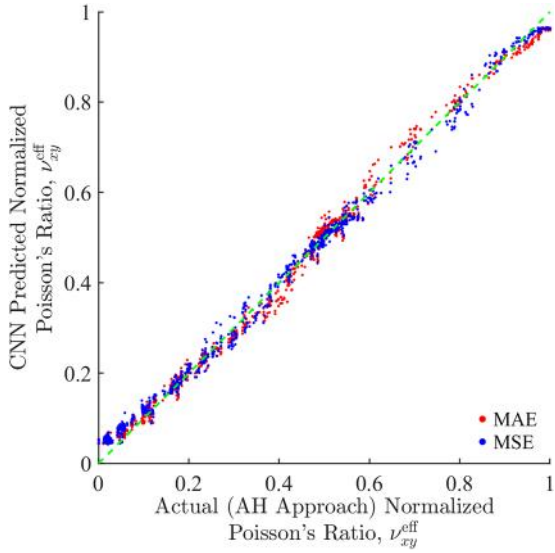


Fig. 11 Comparison between actual and predicted normalized Poisson's ratio ν_{xy}^{eff}

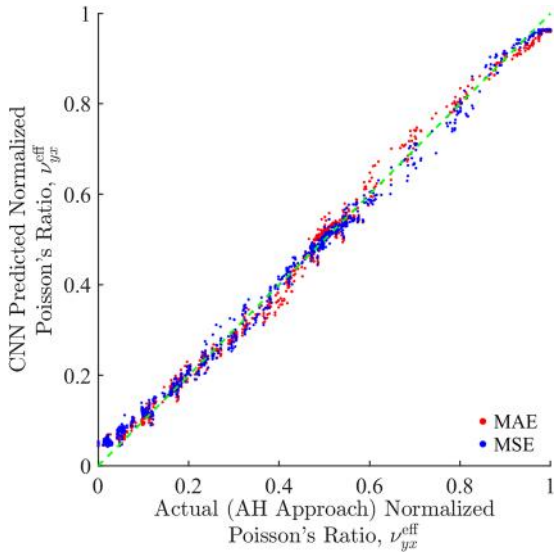


Fig. 12 Comparison between actual and predicted normalized Poisson's ratio ν_{yx}^{eff}

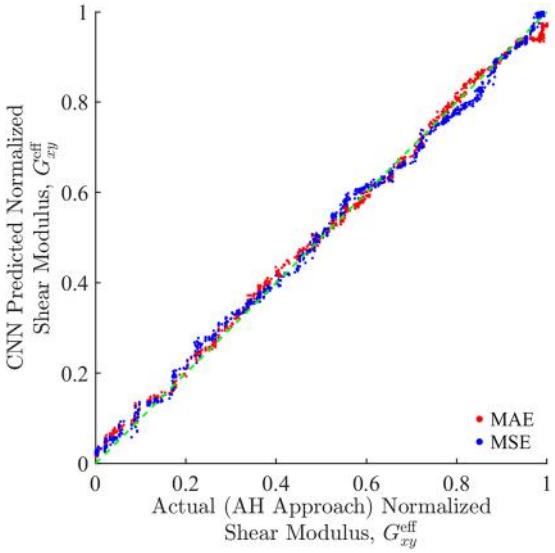


Fig. 13 Comparison between actual and predicted normalized shear modulus G_{xy}^{eff}

5 Conclusions

This study presents a novel framework that integrates CNNs with image processing to predict the homogenized constitutive properties of lattice-unit cells. Instead of relying on explicit geometric parameters, the lattice geometry is represented directly as images, enabling a more general and flexible description of complex lattice topologies. Through image processing, the binary mapping of the unit cell is converted into numerical matrices that are then used for finite-element-based AH. By using the same image-based representation for both CNN learning and finite element analysis, the proposed approach ensures a consistent, unified data format across modeling and ML development.

By training the developed CNN on a dataset of 10,000 RVE images, the network successfully learned the nonlinear relationship between lattice morphology and the corresponding homogenized elastic properties obtained using the AH approach. Following a systematic hyperparameter optimization procedure, the predictive performance of the trained models was evaluated using normalized error metrics to ensure consistent comparison across all material properties. In the initial testing phase on a held-out 10% subset of the original dataset, both training strategies exhibited strong generalization performance, with normalized prediction errors well below 2%. The MAE-trained model achieved a normalized mean absolute error of approximately 1.74% and a normalized mean squared error of about 0.049%, while the MSE-trained model further improved accuracy, attaining a lower normalized mean absolute error of approximately 1.59% and a reduced normalized mean squared error of about 0.041%. To further assess robustness and generalization capability, both final models were subsequently evaluated on an independent dataset consisting of 1000 previously unseen RVE images. Averaged across the five elastic properties, the MAE-trained model yielded a normalized mean absolute error of approximately 1.70%, a normalized mean squared error of 4.48×10^{-4}, and a root mean squared error of about 2.11%, with a high mean coefficient of determination of $R^2 = 0.9949$. The MSE-trained model again demonstrated slightly superior predictive fidelity, achieving a lower normalized mean absolute error of approximately 1.65%, a normalized mean squared error of 4.44×10^{-4}, a root mean squared error of about 2.09%, and a mean R^2 value of 0.9950. In all cases, the CNN-predicted effective Young's moduli, Poisson's ratios, and shear modulus exhibit excellent agreement with the AH-based reference solutions. Moreover, the trained CNN is capable of predicting the effective elastic properties of 1000 RVEs in a fraction of the time required for a single finite element homogenization analysis, highlighting its potential for accelerating lattice design and optimization studies.

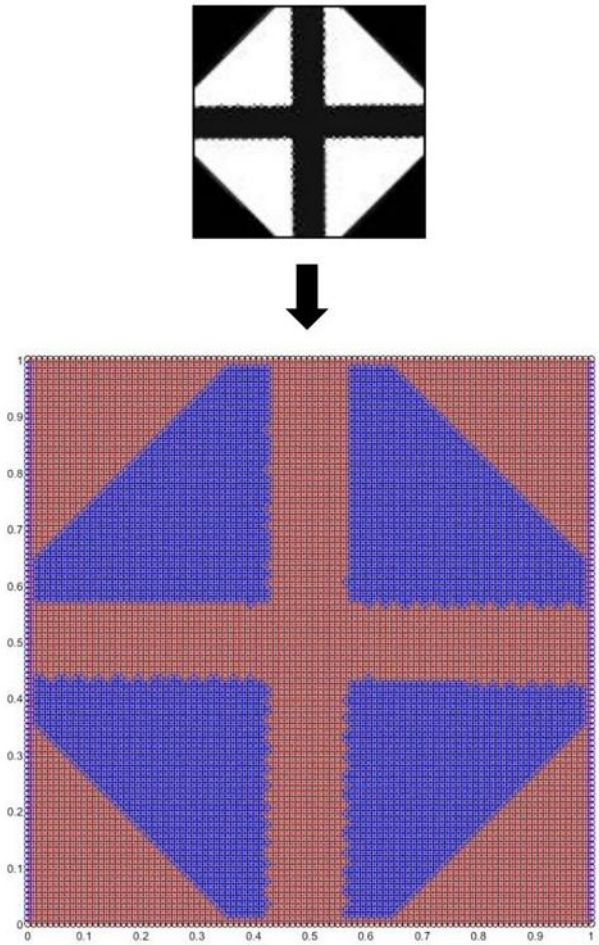


Fig. 14 An image representation of the binary matrix of the optimal RVE from the dataset

Beyond computational efficiency, this work contributes conceptually by demonstrating that image features alone can encode sufficient information to determine effective material properties. This insight opens the door to integrating computer vision and deep learning with classical homogenization theory. The approach also creates opportunities for inverse design, where target material properties could be used to generate new lattice configurations directly from image-based models. Future extensions of this work will focus on several key directions. First, incorporating a mesh correction or image-preprocessing module that can enhance accuracy by identifying and rectifying irregular or corrugated boundaries in binary mapping for accurate finite element model development. Second, the proposed framework can be extended to three-dimensional lattice architectures through 3D binary mapping. When the cross section remains uniform along the thickness, each RVE within the 3D structural member can be treated as a representative volume element, enabling the CNN-based homogenization tool to handle multiple RVE designs in three dimensions. The present study also aims to extend this framework for analyzing complex multiple 3D RVEs.

Acknowledgment

The authors express their gratitude for the financial support provided by Oklahoma State University's New Faculty Startup Funding and partial support from the NASA Oklahoma EPSCoR

Grant (80NSSC22M0029), Karen M. Brown Taminger at NASA Langley Research Center served as the NASA Technical Point of Contact. The authors thank Soumik Dutta, a graduate student in the Mechanical and Aerospace Department at OSU, for helping us in this study. The computing for this project was performed at the High-Performance Computing Center at Oklahoma State University supported in part through the National Science Foundation Grant (OAC-1126330). A special thanks goes to Evan Linde, Research Cyber Infrastructure Analyst, for his assistance in running the code at OSU's High-Performance Computing Center. The manuscript's language and clarity were enhanced with the assistance of AI tools (ChatGPT and Grammarly). The authors performed a comprehensive review and editing of all AI-generated content to guarantee accuracy, originality, and adherence to scholarly standards.

Conflict of Interest

There are no conflicts of interest.

Data Availability Statement

The datasets generated and supporting the findings of this article are obtainable from the corresponding author upon reasonable request.

Nomenclature

f = body force (N/m^3)
r = relative density
t = surface traction (Pa)
v_s = solid Poisson's ratio
$v_{xy}^{eff.}$ = effective Poisson's ratio, transverse strain along Y over longitudinal strain along X
$v_{yx}^{eff.}$ = effective Poisson's ratio, longitudinal strain along X over transverse strain along Y
E_s = solid Young's modulus (Pa)
$E_{xx}^{eff.}$ = effective Young's modulus along X (Pa)
$E_{yy}^{eff.}$ = effective Young's modulus along Y (Pa)
$G_{xy}^{eff.}$ = effective shear modulus in XY plane (Pa)
$\bar{\Omega}$ = homogenized domain
Ω = periodic microstructure domain
Γ_d = displacement boundary
Γ_t = traction boundary

Appendix A: Mean Absolute Error Loss Across All Batch Sizes, Learning Rates (LR), and Optimizers

See Figs. 15–18.

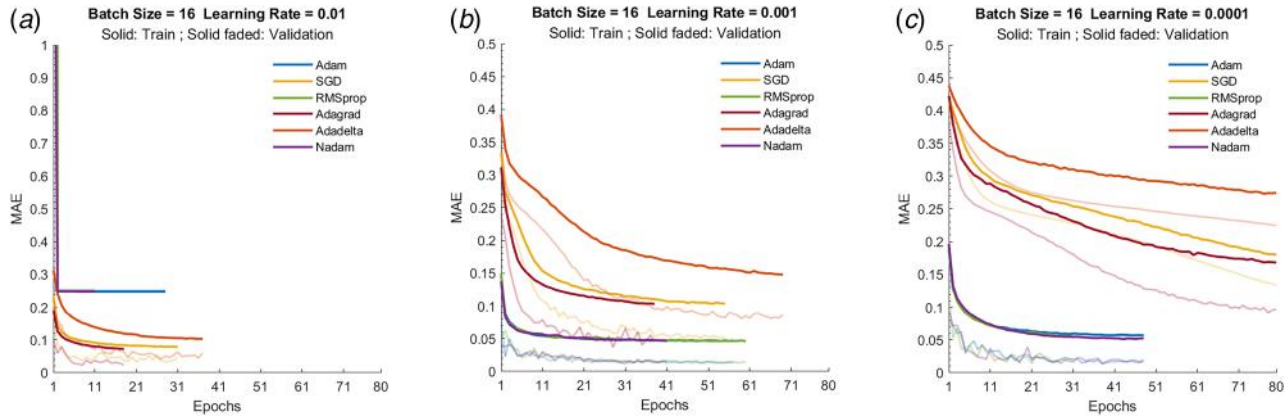


Fig. 15 MAE training and validation loss versus epochs for batch size 16. (*a*) LR: 0.01. (*b*) LR: 0.001. (*c*) LR: 0.0001.

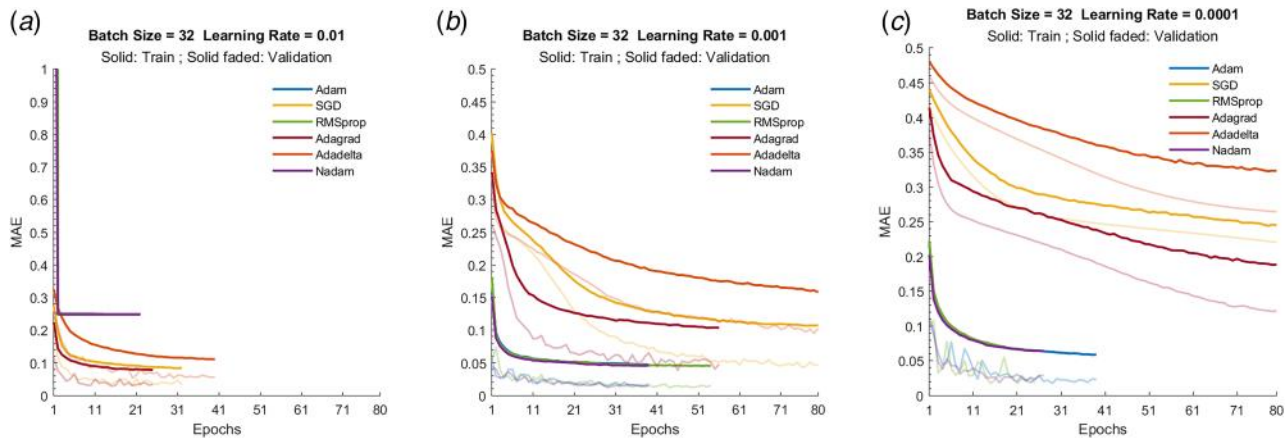


Fig. 16 MAE training and validation loss versus epochs for batch size 32. (*a*) LR: 0.01. (*b*) LR: 0.001. (*c*) LR: 0.0001.

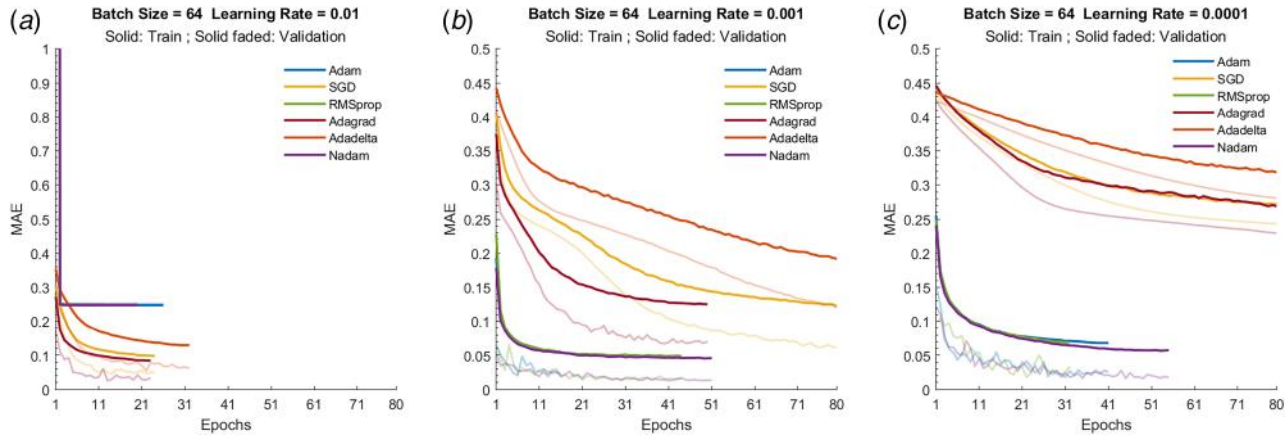


Fig. 17 MAE training and validation loss versus epochs for batch size 64. (*a*) LR: 0.01. (*b*) LR: 0.001. (*c*) LR: 0.0001.

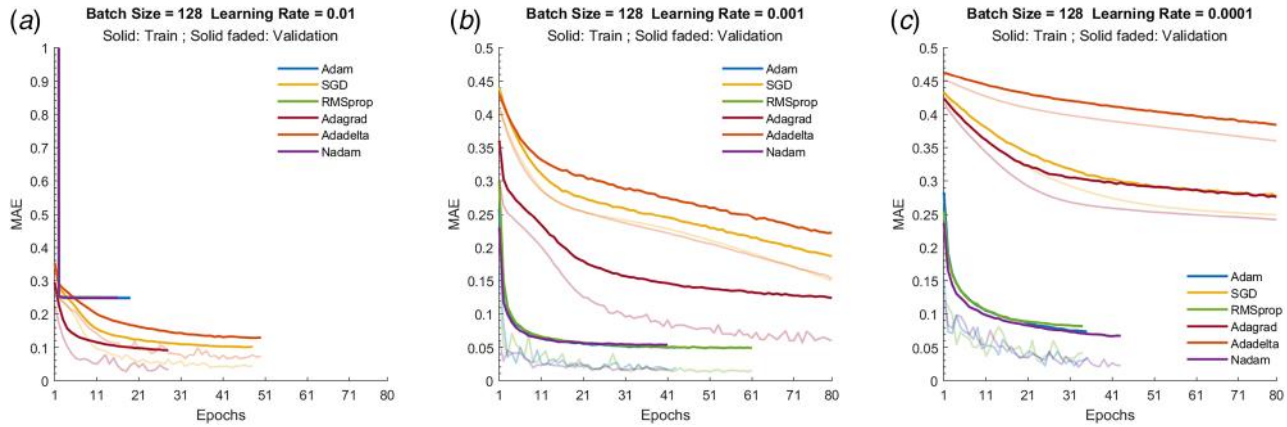


Fig. 18 MAE training and validation loss versus epochs for batch size 128. (*a*) LR: 0.01. (*b*) LR: 0.001. (*c*) LR: 0.0001.

Appendix B: Mean Squared Error Loss Across All Batch Sizes, Learning Rates, and Optimizers

See Figs. 19–22.

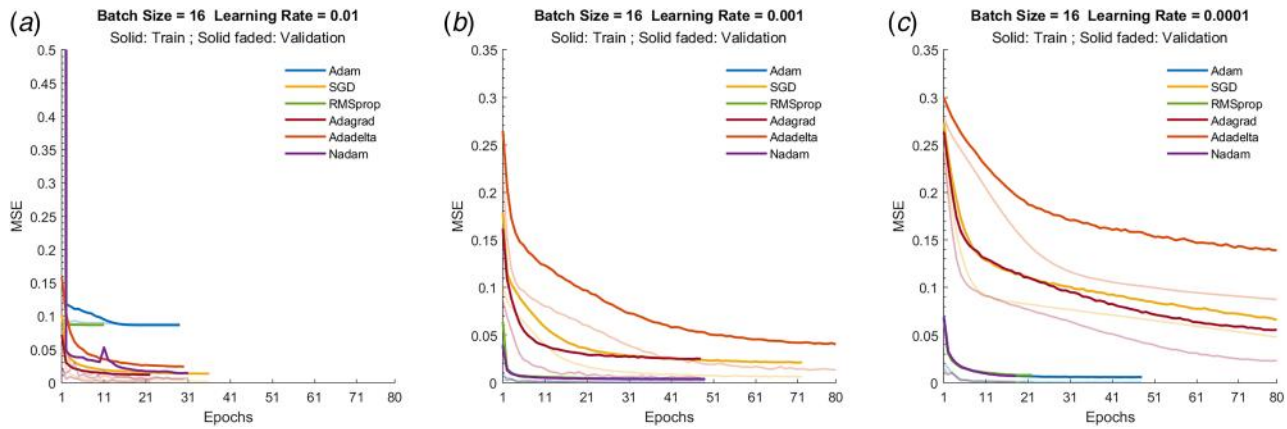


Fig. 19 MSE training and validation loss versus epochs for batch size 16. (*a*) LR: 0.01. (*b*) LR: 0.001. (*c*) LR: 0.0001.

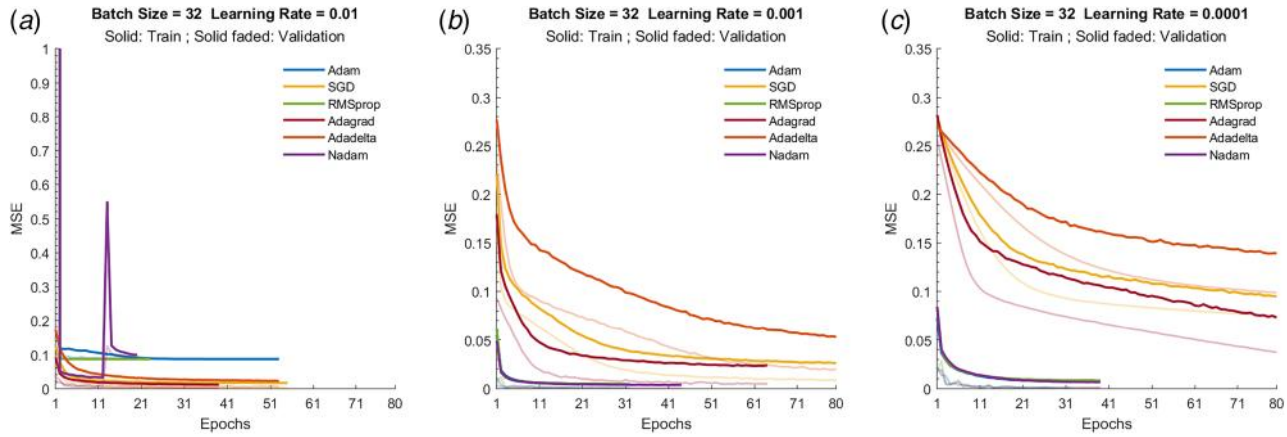


Fig. 20 MSE training and validation loss versus epochs for batch size 32. (*a*) LR: 0.01. (*b*) LR: 0.001. (*c*) LR: 0.0001.

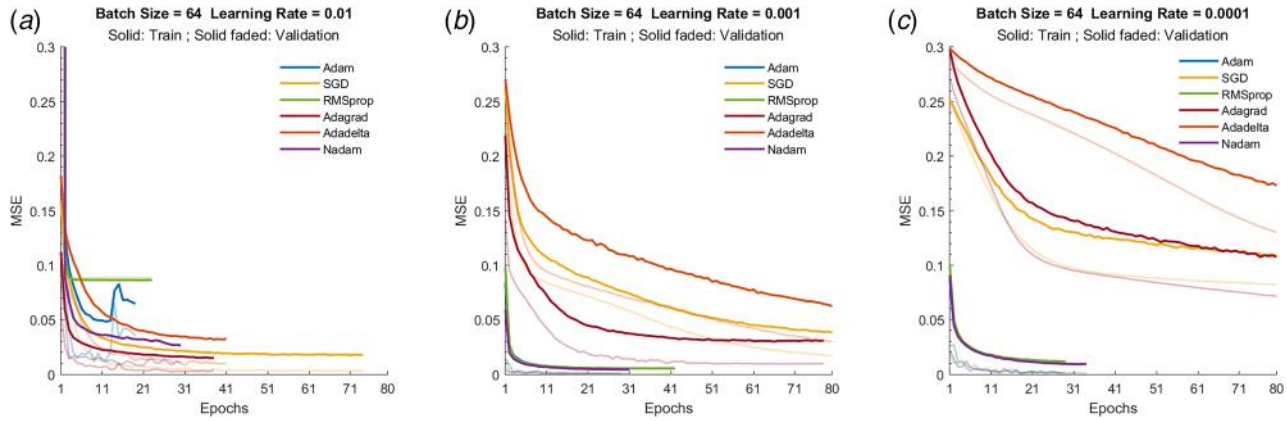


Fig. 21 MSE training and validation loss versus epochs for batch size 64. (*a*) LR: 0.01. (*b*) LR: 0.001. (*c*) LR: 0.0001.

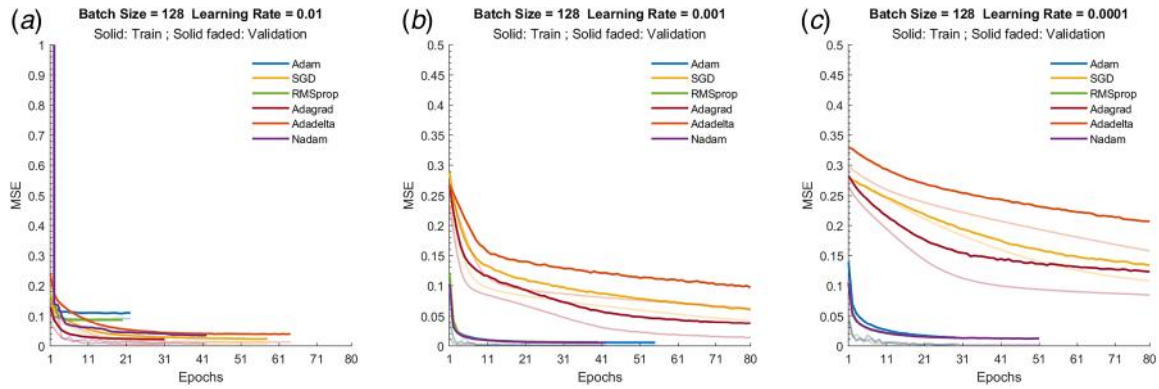


Fig. 22 MSE training and validation loss versus epochs for batch size 128. (*a*) LR: 0.01. (*b*) LR: 0.001. (*c*) LR: 0.0001.

References

[1] Sajjad, U., Rehman, T.-u., Ali, M., Park, C. W., and Yan, W.-M., 2022, "Manufacturing and Potential Applications of Lattice Structures in Thermal Systems: A Comprehensive Review of Recent Advances," Int. J. Heat Mass Transfer, **198**, p. 123352.

[2] Pan, C., Han, Y., and Lu, J., 2020, "Design and Optimization of Lattice Structures: A Review," Appl. Sci., **10**(18), p. 6374.

[3] Wadley, H. N., and Queheillalt, D. T., 2007, "Thermal Applications of Cellular Lattice Structures," Materials Science Forum, **539–543**, pp. 242–247.

[4] Helou, M., and Kara, S., 2018, "Design, Analysis and Manufacturing of Lattice Structures: An Overview," Int. J. Comput. Integr. Manuf., **31**(3), pp. 243–261.

[5] Dubovikov, E., Fomin, D., Guseva, N., Kondakov, I., Kruychkov, E., Mareskin, I., and Shanygin, A., 2019, "Manufacturing Aspects of Creating Low-Curvature Panels for Prospective Civil Aircraft," Aerospace, **6**(2), p., 18.

[6] Zhu, J., Zhou, H., Wang, C., Zhou, L., Yuan, S., and Zhang, W., 2021, "A Review of Topology Optimization for Additive Manufacturing: Status and Challenges," Chin. J. Aeronaut., **34**(1), pp. 91–110.

[7] Arabnejad, S., Johnston, B., Tanzer, M., and Pasini, D., 2017, "Fully Porous 3D Printed Titanium Femoral Stem to Reduce Stress-Shielding Following Total Hip Arthroplasty," J. Orthop. Res., **35**(8), pp. 1774–1783.

[8] Mahdi, M. A., Dutta, S., and Zhao, W., 2024, "Stiffness Tailoring for Improved Buckling of Variable Angle Tow Composite Sandwich Plates," AIAA SCITECH 2024 Forum, Orlando, FL, Jan. 8–12, Paper No. AIAA 2024-0549.

[9] Gibson, I., and Ashby, M. F., 1982, "The Mechanics of Three-Dimensional Cellular Materials," Proc. R. Soc. Lond. A. Math. Phys. Sci., **382**(1782), pp. 43–59.

[10] Masters, I., and Evans, K., 1996, "Models for the Elastic Deformation of Honeycombs," Compos. Struct., **35**(4), pp. 403–422.

[11] Wang, A.-J., and McDowell, D., 2004, "In-Plane Stiffness and Yield Strength of Periodic Metal Honeycombs," ASME J. Eng. Mater. Technol., **126**(2), pp. 137–156.
[12] Somnic, J., and Jo, B. W., 2022, "Homogenization Methods of Lattice Materials," Encyclopedia, **2**(2), pp. 1091–1102.
[13] Hassani, B., and Hinton, E., 1998, "A Review of Homogenization and Topology Optimization I—Homogenization Theory for Media With Periodic Structure," Comput. Struct., **69**(6), pp. 707–717.
[14] Guedes, J. M., and Kikuchi, N., 1990, "Preprocessing and Postprocessing for Materials Based on the Homogenization Method With Adaptive Finite Element Methods," Comput. Methods Appl. Mech. Eng., **83**(2), pp. 143–198.
[15] Hassani, B., and Hinton, E., 1998, "A Review of Homogenization and Topology Opimization II—Analytical and Numerical Solution of Homogenization Equations," Comput. Struct., **69**(6), pp. 719–738.
[16] Hassani, B., and Hinton, E., 1998, "A Review of Homogenization and Topology Optimization III—Topology Optimization Using Optimality Criteria," Comput. Struct., **69**(6), pp. 739–756.
[17] Arabnejad, S., and Pasini, D., 2013, "Mechanical Properties of Lattice Materials via Asymptotic Homogenization and Comparison With Alternative Homogenization Methods," Int. J. Mech. Sci., **77**, pp. 249–262.
[18] Moeini, M., Begon, M., and Lévesque, M., 2022, "Numerical Homogenization of a Linearly Elastic Honeycomb Lattice Structure and Comparison With Analytical and Experimental Results," Mech. Mater., **167**, p. 104210.
[19] Vlădulescu, F., and Constantinescu, D. M., 2020, "Lattice Structure Optimization and Homogenization Through Finite Element Analyses," Proc. Inst. Mech. Eng., Part L: J. Mater. Des. Appl., **234**(12), pp. 1490–1502.
[20] Feyel, F., 1999, "Multiscale FE2 Elastoviscoplastic Analysis of Composite Structures," Comput. Mater. Sci., **16**(1–4), pp. 344–354.
[21] Settgast, C., Hütter, G., Kuna, M., and Abendroth, M., 2020, "A Hybrid Approach to Simulate the Homogenized Irreversible Elastic–Plastic Deformations and Damage of Foams by Neural Networks," Int. J. Plast., **126**, p. 102624.
[22] Alwattar, T. A., and Mian, A., 2019, "Development of an Elastic Material Model for BCC Lattice Cell Structures Using Finite Element Analysis and Neural Networks Approaches," J. Compos. Sci., **3**(2), p. 33.
[23] Gongora, A. E., Friedman, C., Newton, D. K., Yee, T. D., Doorenbos, Z., Giera, B., Duoss, E. B., Han, T. Y. -J., Sullivan, K., and Rodriguez, J. N., 2024, "Accelerating the Design of Lattice Structures Using Machine Learning," Sci. Rep., **14**(1), p., 13703.
[24] Ma, S., Tang, Q., Liu, Y., and Feng, Q., 2022, "Prediction of Mechanical Properties of Three-Dimensional Printed Lattice Structures Through Machine Learning," ASME J. Comput. Inf. Sci. Eng., **22**(3), p.031008.
[25] Hooshmand, M. J., Sakib-Uz-Zaman, C., and Khondoker, M. A. H., 2023, "Machine Learning Algorithms for Predicting Mechanical Stiffness of Lattice Structure-Based Polymer Foam," Materials, **16**(22), p., 7173.
[26] Wang, J., and Panesar, A., 2022, "Machine Learning Based Lattice Generation Method Derived From Topology Optimisation," Addit. Manuf., **60**, p. 103238.
[27] Yu, G., Xiao, L., and Song, W., 2022, "Deep Learning-Based Heterogeneous Strategy for Customizing Responses of Lattice Structures," Int. J. Mech. Sci., **229**, p., 107531.
[28] Ibrahimi, S., D'Andrea, L., Gastaldi, D., Rivolta, M. W., and Vena, P., 2024, "Machine Learning Approaches for the Design of Biomechanically Compatible Bone Tissue Engineering Scaffolds," Comput. Methods Appl. Mech. Eng., **423**, p. 116842.
[29] Sigmund, O., 2000, "Topology Optimization: A Tool for the Tailoring of Structures and Materials," Philos. Trans. R. Soc. Lond. Ser. A, **358**(1765), pp. 211–227.
[30] Minaee, S., Boykov, Y., Porikli, F., Plaza, A., Kehtarnavaz, N., and Terzopoulos, D., 2021, "Image Segmentation Using Deep Learning: A Survey," IEEE Trans. Pattern Anal. Mach. Intell., **44**(7), pp. 3523–3542.
[31] Mahdi, M. A., Crick, C., and Zhao, W., 2024, "Lattice Structure Design Using Machine Learning and Homogenization Approach," ASME 2024 Aerospace Structures, Structural Dynamics, and Materials Conference, Renton, WA, Apr. 29–May 1, Paper No. SSDM2024- 121612.
[32] Takano, N., Ohnishi, Y., Zako, M., and Nishiyabu, K., 2001, "Microstructure-Based Deep-Drawing Simulation of Knitted Fabric Reinforced Thermoplastics by Homogenization Theory," Int. J. Solids Struct., **38**(36–37), pp. 6333–6356.
[33] Andreassen, E., and Andreasen, C. S., 2014, "How to Determine Composite Material Properties Using Numerical Homogenization," Comput. Mater. Sci., **83**, pp. 488–495.
[34] Wang, C., Zhu, J. H., Zhang, W. H., Li, S. Y., and Kong, J., 2018, "Concurrent Topology Optimization Design of Structures and Non-Uniform Parameterized Lattice Microstructures," Struct. Multidisc. Optim., **58**(1), pp. 35–50.
[35] Angra, S., and Ahuja, S., 2017, "Machine Learning and Its Applications: A Review," 2017 International Conference on Big Data Analytics and Computational Intelligence (ICBDAC), Chirala, Andhra Pradesh, India, Mar. 23–25, pp. 57–60.
[36] Lindsay, G. W., 2021, "Convolutional Neural Networks as a Model of the Visual System: Past, Present, and Future," J. Cognit. Neurosci., **33**(10), pp. 2017–2031.
[37] Khurjekar, I. D., Conry, B., Kesler, M. S., Tonks, M. R., Krause, A. R., and Harley, J. B., 2023, "Automated, High-Accuracy Classification of Textured Microstructures Using a Convolutional Neural Network," Front. Mater., **10**, pp. 1–10.
[38] El-Shafie, A. H. A., Zaki, M., and Habib, S. E. D., 2022, "An Efficient Hardware Implementation of CNN-Based Object Trackers for Real-Time Applications," Neural Comput. Appl., **34**(22), pp. 19937–19952.
[39] Stiefel, M., Müller, M., Bachmann, B.-I., Guitar, M. A., Nayak, U. P., and Mücklich, F., 2024, "Enhancing Machine Learning Classification of Microstructures: A Workflow Study on Joining Image Data and Metadata in CNN," MRS Commun., **14**(3), pp. 363–371.
[40] Jia, H., Zhang, J., Ma, K., Qiao, X., Ren, L., and Shi, X., 2024, "Application of Convolutional Neural Networks in Medical Images: A Bibliometric Analysis," Quant. Imaging Med. Surg., **14**(5), pp. 3501–3518.
[41] Liu, M., Shi, J., Li, Z., Li, C., Zhu, J., and Liu, S., 2016, "Towards Better Analysis of Deep Convolutional Neural Networks," IEEE Trans. Vis. Comput. Graph., **23**(1), pp. 91–100.
[42] Jiang, J., Xiong, Y., Zhang, Z., and Rosen, D. W., 2022, "Machine Learning Integrated Design for Additive Manufacturing," J. Intell. Manuf., **33**(4), pp. 1073–1086.
[43] Ladds, M. A., Thompson, A. P., Kadar, J.-P., J Slip, D., P Hocking, D., and G Harcourt, R., 2017, "Super Machine Learning: Improving Accuracy and Reducing Variance of Behaviour Classification From Accelerometry," Anim. Biotelem., **5**(1), p. 8.