

3DSolidFormer for high-resolution full-field prediction in three-dimensional linear elasticity

David Guirguis

dguirguis@smu.edu

Southern Methodist University <https://orcid.org/0000-0001-5341-2437>

Research Article

Keywords: surrogate modeling, three-dimensional linear elasticity, full-field stress prediction, residual U-Net, self-attention, transformers, solid mechanics, topology optimization

Posted Date: October 27th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-11494084/v1>

License:   This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: The authors declare no competing interests.

3DSolidFormer for high-resolution full-field prediction in three-dimensional linear elasticity

David Guirguis

Mechanical Engineering Department, Southern Methodist University, Dallas, TX, USA

Abstract

Repeated finite-element analysis becomes costly when displacement, strain, and stress must be resolved throughout three-dimensional structures on meshes containing millions of elements. In this study, 3DSolidFormer is developed for simultaneous prediction of these fields from material distributions, applied forces, and displacement constraints. Global structural dependencies and field-specific local interactions are learned through a transformer-enhanced architecture, while prescribed zero-displacement constraints are satisfied by construction. The accompanying dataset contains 75,114 finite-element solutions for three-dimensional domains discretized into 2.05 to 3.07 million voxels, covering fully solid and topology-optimized structures under small-strain linear elasticity. Across 7,488 high-resolution test cases, median case-level relative root-mean-square errors are 0.1172 for displacement, 0.1475 for equivalent strain, and 0.1448 for von Mises stress. Mean Percentage Mean Absolute Error (PMAE) values are 2.62% for displacement magnitude, 0.204% for equivalent strain, and 0.198% for von Mises stress. Direct scalar-field prediction reduces median PMAE by 76.6% for strain and 76.9% for stress relative to recovery from predicted displacement. All three fields are returned at the high-resolution mesh nodes in a single forward pass. The dataset provides reference solutions for benchmarking full-field surrogates and developing learning-based structural optimization methods.

Keywords: surrogate modeling, three-dimensional linear elasticity, full-field stress prediction, residual U-Net, self-attention, transformers, solid mechanics, topology optimization.

1. Introduction

Structural optimization repeatedly evaluates how material layout affects deformation and stress. Resolving these responses throughout a three-dimensional structure makes repeated finite-element analysis (FEA) costly. Surrogate models reduce this burden by learning the relationship between inputs and responses [1]. Classical methods, including Gaussian-process regression, polynomial response surfaces, and radial-basis-function interpolation, commonly predict scalar quantities from design variables. These methods can be data-efficient, although training becomes more costly as input dimension and dataset size increase [1].

Model-order reduction retains the governing equations while reducing the solution dimension. Proper orthogonal decomposition (POD) constructs a trial space from full-order snapshots [2], whereas proper generalized decomposition (PGD) uses separated spatial, temporal, and parametric representations [3]. Their efficiency depends on how many modes or separated terms are needed to represent the solution. Changing topology and strongly varying fields can require many modes [2]. Physics-informed neural networks incorporate governing-equation residuals

into training [4], although balancing these residuals with data errors can make optimization difficult [5].

Deep-learning surrogates learn spatial representations directly from data. In solid mechanics, Nie et al. used convolutional neural networks (CNNs) to predict two-dimensional stress fields [6], and StressGAN used conditional generative modeling across varying configurations [7]. MeshGraphNets represents interactions between mesh nodes through message passing [8]. Neural operators instead learn mappings between functions: DeepONet separates input-function encoding from evaluation at output coordinates [9], including for elastoplastic stress prediction under varied geometries and loads [10].

Attention-based operator models learn relationships between spatial locations. Cao introduced Fourier- and Galerkin-type transformers for partial differential equations from an operator-approximation perspective [11]. OFormer combines self-attention, cross-attention, and point-wise multilayer perceptrons to separate input sampling locations from output query points [12].

Three-dimensional mechanics prediction must capture both overall deformation and local strain and stress variations near loads, supports, and narrow members. Available datasets address different parts of this problem. Mechanical MNIST contains 70,000 two-dimensional simulations with displacement histories and energy-related quantities [13]. SimJEB provides 381 three-dimensional bracket designs with displacement and von Mises stress [14], while DeepJEB extends this family to over 2,000 designs with high-resolution nodal fields [15]. The MeshGraphNets Deforming Plate dataset addresses dynamic hyperelastic deformation [8]. The present study considers joint displacement, equivalent strain, and von Mises stress prediction for solid and topology-optimized structures under small-strain linear elasticity.

These challenges motivate the development of multiresolution datasets alongside a systematic assessment of convolutional feature extraction, attention mechanisms, and field-specific prediction heads. Building on this motivation, this paper introduces 3DSolidFormer, a transformer-enhanced architecture for full-field prediction in low and high-resolution three-dimensional solid mechanics. The architecture combines a residual 3D U-Net backbone with global-context attention and task-specific output mappings to predict nodal displacement, equivalent strain, and von Mises stress from density fields, applied loads, and displacement constraints. Training uses finite-element-generated labels and a multicomponent supervised objective, with exact enforcement of prescribed zero-displacement constraints. Controlled architecture ablations examine the contributions of network width, and output-head design, while matched low and high-resolution datasets support investigation of resolution-dependent performance. The accompanying dataset provides a benchmark for reproducible evaluation of three-dimensional mechanics surrogates. Particular emphasis is placed on accuracy across different response magnitudes and transfer to finer spatial discretizations, providing a basis for future surrogate-assisted topology optimization.

The rest of the paper is organized as follows. Section 2 describes the dataset, finite-element formulation, network architecture, and training objective. Section 3 presents the architecture comparisons and high-resolution results, including spatial accuracy and computational performance. Section 4 summarizes the findings and directions for further study.

2. Methodology

2.1. Dataset construction

The dataset contains three-dimensional structures with different domain shapes, material layouts, loads, and displacement constraints. Each source design is analyzed as a topology-optimized structure and, where available, as a fully solid counterpart under the same loads

and supports. These matched cases permit comparison of material distributions under common boundary conditions. Nodal displacement, equivalent strain, and von Mises stress are obtained by finite-element analysis for training and evaluation.

2.1.1. Domains, supports, and loads

The domain dimensions, support definitions, load locations and directions, and optimized density fields were inherited from NITO-3D [16]. The source dataset contains 210 configuration families defined by seven domain grids, five support-component patterns, and six support-location patterns, with randomized realizations within each family. The high-resolution element grids, in physical x-y-z order, are $480 \times 80 \times 80$, $128 \times 128 \times 128$, $240 \times 160 \times 80$, $160 \times 160 \times 80$, $480 \times 160 \times 40$, $320 \times 160 \times 60$, and $256 \times 128 \times 64$. Each grid contains 2,048,000 to 3,072,000 voxel elements, including solid and void regions. Different domain aspect ratios, support arrangements, volume fractions, and topologies are represented.

Each source realization has three to five supports on prescribed boundary planes, constraining one to three Cartesian components, and one exterior-surface load with total magnitude 1 N. The load acts along a principal axis with probability 0.10 per axis, a principal plane with probability 0.15 per plane, or all three axes with probability 0.25. Loads and supports occupy clipped cubic patches of nodes. On the high-resolution mesh, the patch radius is four nodes, giving up to $9 \times 9 \times 9$ nodes per patch. A source identity specifies the domain, optimized topology, supports, and load. Its fully solid counterpart has material throughout the same domain. Available counterparts are grouped by this identity when the data are partitioned. Domain and support families recur across the partitions. Dataset composition and the spatial distributions of loads and supports are shown in Figs. 1 and 2.

2.1.2. Finite-element formulation

The mechanical model uses small-strain, isotropic linear elasticity on a structured mesh of trilinear hexahedral elements (Q1) [17]. For a displacement field $\mathbf{u}$, the infinitesimal strain is

$$\boldsymbol{\varepsilon}(\mathbf{u}) = \frac{1}{2} [\nabla \mathbf{u} + (\nabla \mathbf{u})^\top] \quad (1)$$

The baseline constitutive stress and Lamé parameters are

$$\begin{aligned} \sigma_0 &= 2\mu_0 \boldsymbol{\varepsilon} + \lambda_0 \text{tr}(\boldsymbol{\varepsilon}) \mathbf{I}, \\ \mu_0 &= \frac{E_0}{2(1+\nu)}, \quad \lambda_0 = \frac{E_0 \nu}{(1+\nu)(1-2\nu)}. \end{aligned} \quad (2)$$

The binary material distribution is represented by assigning a small residual stiffness to void elements to regularize the finite-element system:

$$\mathbb{C}_e = \begin{cases} \eta \mathbb{C}_0, & \rho_e = 0 \quad (\text{void}), \\ (1+\eta) \mathbb{C}_0, & \rho_e = 1 \quad (\text{solid}), \end{cases} \quad \sigma_e = \mathbb{C}_e : \boldsymbol{\varepsilon}_e \quad (3)$$

Here, density indicator $\rho_e \in \{0,1\}$ identifies void and solid elements, $\mathbb{C}_0$ is the baseline elasticity tensor associated with E_0 and ν , and $\eta = 10^{-6}$. The binary voxel geometry is used directly. The solid stiffness is $(1+\eta)$ times the baseline value, and stresses use the constitutive tensor of each element.

The reference scales are $E_0 = 1$ Pa and a source-grid spacing of $h_{LR} = 1$ m, with Poisson ratio $\nu = 0.30$. Forces are expressed in N and the density indicator is dimensionless. The benchmark is defined using these reference values, which do not represent a calibrated engineering material. Each Q1 element is integrated using eight Gauss points ($2 \times 2 \times 2$). The weak void material is

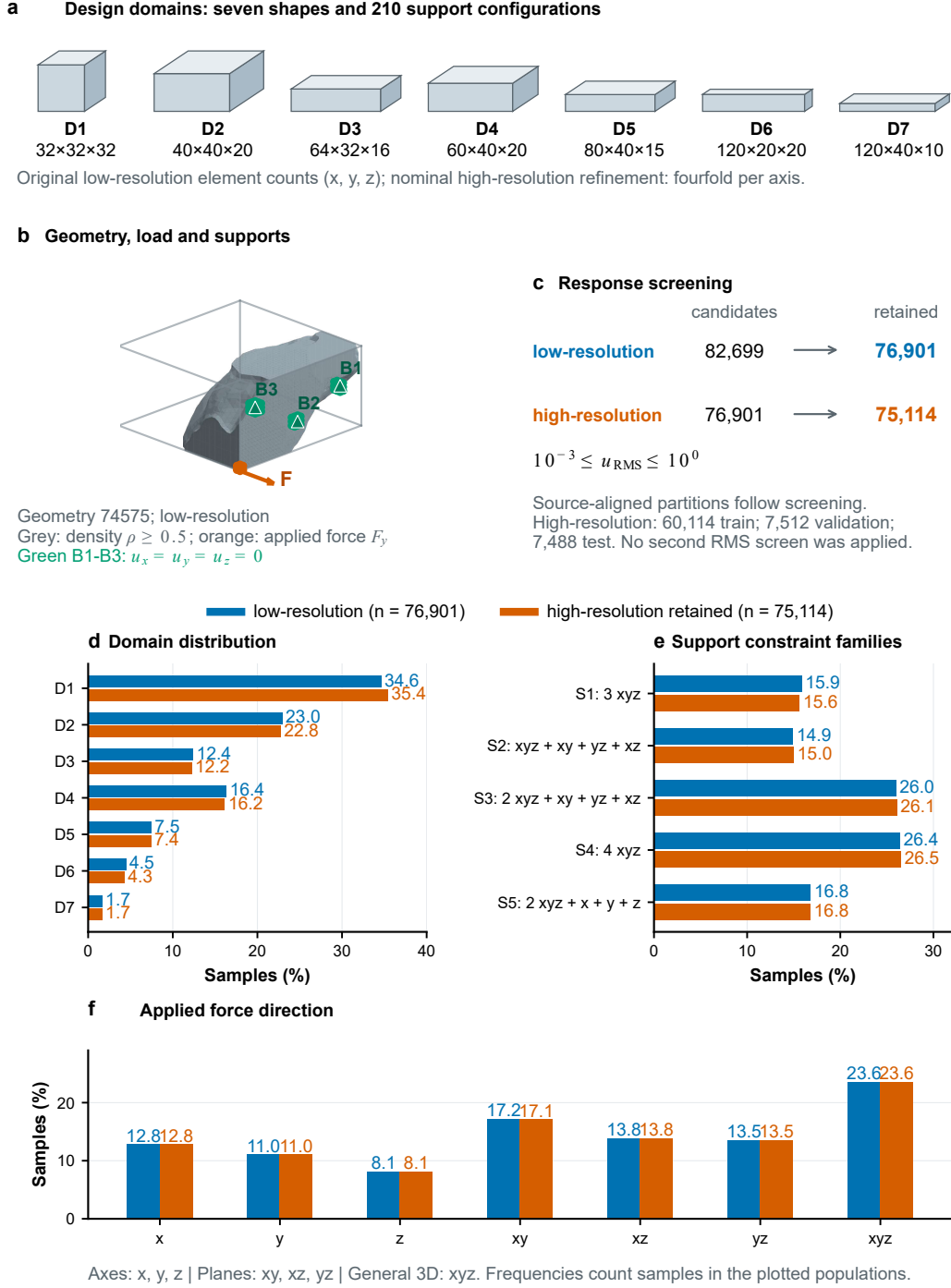


Fig. 1. Dataset construction and composition. (a) The seven source-domain grids and 210 domain/support families. High-resolution meshes subdivide each source element fourfold along every axis. (b) Representative topology, applied load, and supports, with density shown at isovalue 0.5. (c) Displacement-RMS screening retains 76,901 of 82,699 low-resolution candidates and 75,114 of 76,901 high-resolution candidates within the interval 10^{-3} to 1. The source-aligned high-resolution split assigns the retained samples to 60,114 training, 7,512 validation, and 7,488 test cases. (d-f) Domain, support-pattern, and load-direction distributions of the retained populations. Domain schematics are not drawn to a common physical scale.

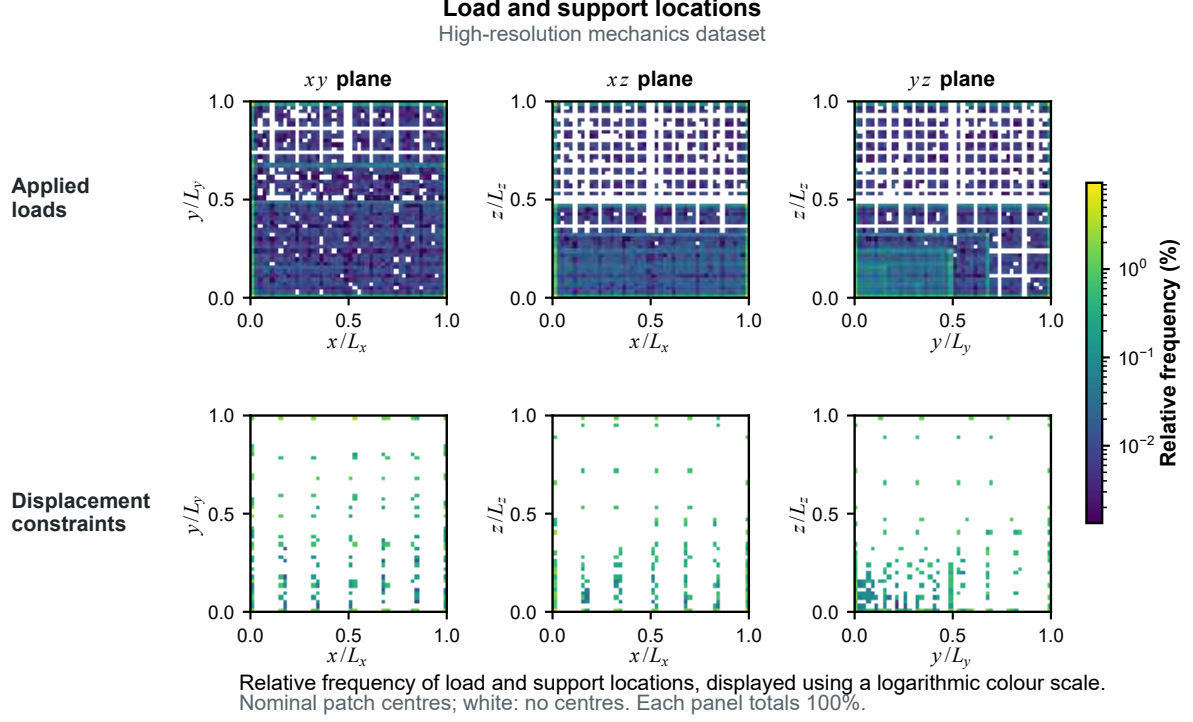


Fig. 2. Spatial distributions of source loading and supports in normalized domain coordinates. The projected histograms use 48×48 bins. Color indicates relative frequency on the plotted logarithmic scale. These distributions describe the retained high-resolution population. Regular white bands arise from binning discrete nodal coordinates and do not represent unsampled spatial regions.

included in the finite-element solution and training targets. The masks used in training and evaluation are defined in Section 2.3. For homogeneous displacement supports, the weak problem is to find $u_h \in \mathcal{V}_0$ such that

$$\sum_e \int_{\Omega_e} \varepsilon(\mathbf{v}_h) : \mathbb{C}_e : \varepsilon(\mathbf{u}_h) d\Omega = \mathbf{v}^T \mathbf{f}, \forall \mathbf{v}_h \in \mathcal{V}_0 \quad (4)$$

Here $\mathcal{V}_0$ contains admissible Q1 displacement fields with zero constrained components. $\mathbf{v}$ is the corresponding nodal vector. Partitioning the discrete vector into free and constrained degrees of freedom gives

$$\mathbf{K}_{ff}(\rho) \mathbf{u}_f = \mathbf{f}_f, \mathbf{u}_c = \mathbf{0} \quad (5)$$

The full vector $\mathbf{K} \mathbf{u} - \mathbf{f}$ contains support reactions on constrained components. The total applied load is distributed uniformly over its clipped cubic patch of nodes. If $\mathcal{P}$ denotes the patch and F its total applied force, the nodal assignment is

$$\mathbf{f}_a = \frac{\mathbf{F}}{|\mathcal{P}|} (a \in \mathcal{P}), \sum_{a \in \mathcal{P}} \mathbf{f}_a = \mathbf{F} \quad (6)$$

Equivalent strain and von Mises stress are evaluated at quadrature points. The deviatoric strain and stress tensors are defined as

$$\varepsilon_{dev} = \varepsilon - \frac{1}{3} \text{tr}(\varepsilon) \mathbf{I}, \quad \sigma_{dev} = \sigma - \frac{1}{3} \text{tr}(\sigma) \mathbf{I} \quad (7)$$

The scalar invariants are

$$\varepsilon_{eq} = \sqrt{\frac{2}{3} \varepsilon_{dev} : \varepsilon_{dev}} = \sqrt{\frac{2}{3} \left[\varepsilon_{dev,xx}^2 + \varepsilon_{dev,yy}^2 + \varepsilon_{dev,zz}^2 + 2(\varepsilon_{dev,xy}^2 + \varepsilon_{dev,xz}^2 + \varepsilon_{dev,yz}^2) \right]} \quad (8)$$

$$\sigma_{vm} = \sqrt{\frac{3}{2} \sigma_{dev} : \sigma_{dev}}, \quad (9)$$

$$\sigma_{vm}^2 = \frac{(\sigma_{xx} - \sigma_{yy})^2 + (\sigma_{yy} - \sigma_{zz})^2 + (\sigma_{zz} - \sigma_{xx})^2}{2} + 3(\sigma_{xy}^2 + \sigma_{xz}^2 + \sigma_{yz}^2)$$

The shear entries in Eqs. (8) and (9) are tensor components. Each scalar is first averaged over the eight quadrature points and then transferred to each node by averaging the scalar values of its incident elements:

$$\bar{q}_e = \frac{1}{8} \sum_{g=1}^8 q_{e,g}, \quad q_a = \frac{1}{|E(a)|} \sum_{e \in E(a)} \bar{q}_e, \quad q \in \{\varepsilon_{eq}, \sigma_{vm}\} \quad (10)$$

where $E(a)$ is the set of elements incident on node a . The scalar invariants are computed before averaging, which generally differs from computing an invariant from an averaged tensor. Nodal density is obtained by averaging adjacent element densities. The support channels are component-wise binary indicators. For homogeneous isotropic material, the quadrature-point invariants satisfy $\sigma_{vm} = 3\mu\varepsilon_{eq}$. For fully solid variants, all incident elements use the solid shear modulus $(1 + \eta)\mu_0$, so the recovered fields satisfy $\sigma_{vm} = 3(1 + \eta)\mu_0\varepsilon_{eq}$ on the evaluation mask. This relation defines a constitutive-consistency condition for the scalar predictions in fully solid cases. In topology-optimized cases, nodes near solid-void interfaces receive contributions from elements with different moduli, so a single conversion factor cannot relate the two recovered scalar fields throughout the structure. Both scalar fields are predicted directly by separate output heads and compared with recovery from predicted displacement in Section 3.6.

2.1.3. Mesh generation and finite-element solution

The high-resolution meshes are constructed by subdividing each source-grid element into $4 \times 4 \times 4$ elements. The element count therefore increases by a factor of 64, and voxel spacing decreases from $h_{LR} = 1$ to $h_{HR} = 0.25$ in the normalized coordinates. Each child element inherits the material indicator of its parent, preserving the binary geometry. Load and support patches are mapped to the new mesh with the same physical extent and total applied force. This procedure provides a finer discretization of the same mechanical problem.

Coarse-grid solutions used to initialize the iterative solver were computed in single precision using NVIDIA Warp on CUDA [18]. High-resolution analyses employed a matrix-free formulation in double precision [17, 19]. Both solvers used the diagonally preconditioned conjugate-gradient method with a maximum of 30,000 iterations per attempt. A conjugate-residual fallback was available for the coarse-grid solves [20]. For the high-resolution solver, the corresponding coarse-grid displacement field was trilinearly interpolated to initialize the iterative solution. Convergence was assessed using the relative equilibrium residual over unconstrained degrees of freedom:

$$\frac{\|\mathbf{P}_f (\mathbf{K}\mathbf{u} - \mathbf{f})\|_2}{\|\mathbf{P}_f \mathbf{f}\|_2} \leq 10^{-4} \quad (11)$$

where $\mathbf{P}_f$ selects valid, unconstrained displacement components. Constrained degrees of freedom and tensor padding are excluded. A nonzero free-load norm is required in Eq. (11).

Samples were retained only after the finite-element solution satisfied Eq. (11). Equivalent strain and von Mises stress targets were derived from the accepted displacement fields. Recovered strain and stress can remain sensitive to mesh spacing near loads and supports. The evaluation mask in Section 2.3 excludes nodes carrying loads or constraints, but not their surrounding neighborhoods.

2.1.4. Response filtering and dataset partitioning

The displacement responses span several orders of magnitude. Absolute errors can therefore be dominated by large-amplitude responses, while relative errors become sensitive to small absolute deviations as the FEM response approaches zero. Samples were screened by root-mean-square (RMS) displacement to limit the response range:

$$RMS_{u,n} = \sqrt{\left[\frac{1}{3N_n} \sum_{i=1}^{N_n} \sum_{c=1}^3 u_{n,i,c}^2 \right]} \quad (12)$$

where N_n denotes the number of valid nodes in sample n , and $u_{n,i,c}$ is displacement component c at node i . Padded nodes were excluded. Samples satisfying $10^{-3} \leq RMS_{u,n} \leq 1$ were retained to define the response range of the study. These empirical bounds are independent of the finite-element convergence criterion in Eq. (11).

The high-resolution displacement-RMS screen retained 75,114 of 76,901 candidate samples within the interval 10^{-3} to 1. The retained cases were divided into 60,114 training, 7,512 validation, and 7,488 test cases. No second RMS screen was applied after partitioning. Available fully solid and topology-optimized variants were grouped by source identity. Each source was assigned to the same partition at both resolutions, keeping training, validation, and test identities separate.

2.2. Surrogate model architecture

2.2.1. Input representation and residual U-Net backbone

The channel-first input and target tensors are

$$\mathbf{X} \in \mathbb{R}^{N_b \times 7 \times D \times H \times W}, \mathbf{Y} \in \mathbb{R}^{N_b \times 5 \times D \times H \times W}, \quad (13)$$

$$X = (\rho, f_x, f_y, f_z, c_x, c_y, c_z), Y = (u_x, u_y, u_z, \varepsilon_{eq}, \sigma_{vm})$$

where N_b is the batch size and $D \times H \times W$ is the padded spatial shape. Each spatial dimension is extended to the next multiple of 16 by adding zeros after the last valid grid index. A valid-node mask retains the original extent, and padded values are excluded from losses and metrics. The backbone is a residual 3D U-Net with four encoder stages and a bottleneck [21, 22]. For base width b , the encoder widths are $b, 2b, 4b$, and $8b$. The bottleneck width is $16b$, capped at 256. The reference model uses $b = 16$ and the reduced-width variant uses $b = 8$. Each residual block contains two $3 \times 3 \times 3$ convolutions, Group Normalization with at most eight groups chosen to divide the channel count [23], a dropout rate of 0.1 (0.2 at the bottleneck), and an identity or $1 \times 1 \times 1$ residual projection. Downsampling uses a stride-two convolution. Upsampling uses trilinear interpolation, a channel projection, concatenation with the corresponding encoder features, and a residual block:

$$h_1 = \phi_a(\text{GN}(W_1 * x)), h_2 = \text{GN}(W_2 * \text{Drop}(h_1)), \quad (14)$$

$$R(x) = \phi_a(h_2 + P * x)$$

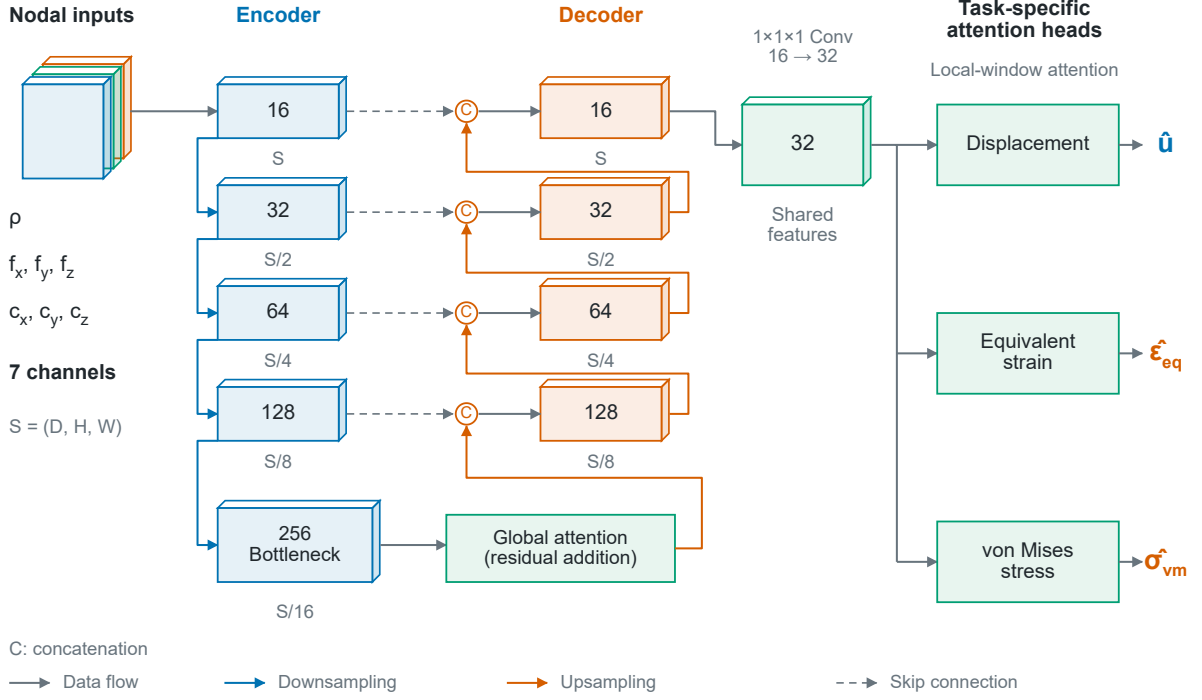


Fig. 3. Architecture of 3DSolidFormer with task-specific attention heads. Seven nodal channels represent density, forces, and displacement constraints. A residual 3D U-Net combines features across spatial scales through skip connections, with global attention at the bottleneck. The shared feature field feeds three independent attention heads for displacement, equivalent strain, and von Mises stress. Each head includes a channel adapter, local-window attention, and a field projection. The shared-attention variant instead uses one attention block before the three field projections. Prescribed displacement components are set to zero, and Softplus ensures nonnegative scalar outputs. Encoder and decoder boxes denote residual Conv3D stages. Numbers give their output channels, with spatial dimensions below each stage. $S = (D, H, W)$ denotes the padded spatial shape, and C denotes concatenation.

where the symbol $*$ denotes convolution. P is an identity or a channel-matching projection. The residual blocks use GELU with a trainable slope [24, 25]. Each activation site has one trainable scalar a , initialized to one, and a fixed multiplier $s_a = 1$:

$$\phi_a(z) = \text{GELU}(s_a a z), s_a = 1, a_0 = 1 \quad (15)$$

The trainable activation slope is optimized without a slope-recovery regularizer. The task-specific-attention configuration used for high-resolution prediction is shown in Fig. 3. Global attention connects bottleneck features across the domain, while local attention processes the features used by the output heads.

2.2.2. Global context

The bottleneck of the reference model is adaptively average-pooled to at most eight cells along each axis, converted to a sequence of feature tokens, and combined with a learned three-dimensional positional embedding. The positional embedding is defined on an $8 \times 8 \times 8$ grid and trilinearly interpolated without corner alignment to the actual pooled token-grid shape. Two pre-normalized Transformer encoder layers [26] use eight attention heads, an MLP expansion ratio of four, no dropout in the attention or MLP sublayers, and fixed GELU in the feed-forward sublayers [24]. For query, key, and value matrices Q , K , and V , and head dimension d_h , the attention operation is

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_h}}\right)V \quad (16)$$

The normalized token field is reshaped, trilinearly interpolated to the bottleneck resolution, passed through a zero-initialized $1 \times 1 \times 1$ projection, and added to the original bottleneck features.

2.2.3. Output heads

In the shared-attention configuration, the U-Net output is projected to 32 feature channels. Nonoverlapping $4 \times 4 \times 4$ windows are processed by one local-window self-attention block with four heads, an MLP expansion ratio of two, a dropout rate of 0.1, learned within-window positional embeddings, final Layer Normalization, and locally adaptive GELU [25]. Three $1 \times 1 \times 1$ projections then produce the displacement and scalar outputs:

$$\begin{aligned} Z_r &= \mathcal{T}_w(Z), u^0 = \Pi_u Z_r, \\ \hat{\varepsilon}_{eq} &= g_\beta(\Pi_\varepsilon Z_r), \hat{\sigma}_{vm} = g_\beta(\Pi_\sigma Z_r) \end{aligned} \quad (17)$$

where $\mathcal{T}_w$ denotes local-window attention followed by normalization, and

$$g_\beta(z) = \frac{1}{\beta} \log(1 + \exp(\beta z)), \beta = 5 \quad (18)$$

Softplus is evaluated by its numerically stable implementation. Scalar outputs are restricted to nonnegative values, but the constitutive relation is not enforced by this activation.

In the task-specific-attention configuration, the 32-channel U-Net features are retained, and a separate channel adapter and local-window attention block are used for each field. Their window size, depth, head count, MLP ratio, dropout, positional encoding, normalization, and activation match the shared-attention configuration. The task-head outputs are denoted by y_f^0 . For displacement, $y_u^0 = u^0$ is the raw field before support projection. The scalar outputs require no such projection:

$$\begin{aligned} Z_r^{(f)} &= \mathcal{T}_{w,f}(A_f Z), y_f^0 = \psi_f(\Pi_f Z_r^{(f)}), \\ \psi_u(z) &= z, \psi_\varepsilon(z) = \psi_\sigma(z) = g_\beta(z) \end{aligned} \quad (19)$$

In the CNN output head configuration, the U-Net directly predicts three raw displacement channels. Each scalar output head receives the seven inputs concatenated with a stop-gradient copy of that displacement. The sequence consists of a $3 \times 3 \times 3$ convolution from 10 to 32 channels, fixed GELU, a second convolution with the same kernel size that refines the 32-channel feature representation while preserving its spatial dimensions, fixed GELU, and a $1 \times 1 \times 1$ projection to one scalar channel, followed by Softplus. For $s \in \{\varepsilon_{eq}, \sigma_{vm}\}$,

$$\begin{aligned} h_{s,1} &= \text{GELU}\left(W_{s,1}^{(3)} * [X, \text{stopgrad}(u^0)]\right), \\ h_{s,2} &= \text{GELU}\left(W_{s,2}^{(3)} * h_{s,1}\right), \hat{y}_s = g_\beta(W_{s,3}^{(1)} * h_{s,2}) \end{aligned} \quad (20)$$

The superscript gives the cubic kernel width. The scalar heads, their inputs, and their gradient paths are changed in this variant, and local-window attention is removed from the displacement output. The comparison therefore assesses the complete output-head design. The CNN output head configuration retains the global-context self-attention encoder.

2.3. Loss function and boundary-condition enforcement

2.3.1. Domain masking and treatment of constrained nodes

Displacement, equivalent strain, and von Mises stress in the loss functions and evaluation metrics denote the nondimensional values u/L_0 , ε_{eq} , σ_{vm}/E_0 , respectively. The fixed reference scales are $L_0 = 1 \text{ m}$ and $E_0 = 1 \text{ Pa}$. They leave the stored numerical values unchanged and are distinct from the per-sample RMS normalization below. All loss terms, coefficients, and numerical floors in Eqs. (22) to (33) are dimensionless under this convention.

Let $V_{n,i}$ be the binary indicator of a valid node and let $B_{n,i}$ be one when any load or support channel exceeds 10^{-12} in magnitude. The evaluation mask is $E_{n,i} = V_{n,i}(1 - B_{n,i})$. For component c , define the fixed mask $F_{n,i,c} = V_{n,i}\mathbf{1}(|c_{n,i,c}| > 10^{-12})$ and free mask $Q_{n,i,c} = V_{n,i} - F_{n,i,c}$. The raw displacement is projected before field losses and metrics:

$$\hat{u}_{n,i,c} = Q_{n,i,c}u_{n,i,c}^0, M_{n,i,c}^u = Q_{n,i,c}E_{n,i}, M_{n,i}^s = E_{n,i} \quad (21)$$

Scalar losses use the same evaluation mask without density reweighting. Because all support-marker nodes are removed by the evaluation mask, the hard projection does not alter the reported direct-field losses or metrics. The boundary-condition penalty regularizes raw values at fixed components, while the prescribed zero displacements are enforced by projection at inference. Values at loaded or supported nodes and support reactions are therefore not assessed by these field errors. The raw prediction at fixed components is separately penalized by

$$\mathcal{L}_{BC} = \frac{\sum_{n,i,c} F_{n,i,c} (u_{n,i,c}^0)^2}{\max(\sum_{n,i,c} F_{n,i,c}, 1)} \quad (22)$$

This penalty acts on u^0 , not the projected value, which is identically zero at fixed components. Targets are normalized within the relative losses for each sample, without global standardization. The BC penalty uses raw displacement divided by L_0 and is therefore dimensionless. Valid evaluation nodes in the weak-material region are included in displacement training and in the scalar losses and their normalization. Reported scalar metrics are restricted to nodes with density at least 0.5, excluding the weak-void predictions that contribute to training.

2.3.2. Relative field losses

Let $f \in \{u, \varepsilon_{eq}, \sigma_{vm}\}$ denote a field group, and let j enumerate its scalar entries (node-component pairs for displacement, nodes for the scalar fields). For each sample,

$$s_{n,f} = \sqrt{\max\left(\frac{\sum_j M_{n,j,f} y_{n,j,f}^2}{\max(\sum_j M_{n,j,f}, 1)}, \epsilon_r^2\right)}, \epsilon_r = 10^{-6} \quad (23)$$

Target RMS is used only to calculate losses and errors. Only material, load, and constraint fields are supplied to the network. The mask is binary in these experiments.

Let $\mathcal{A}_f$ be the samples with at least one active entry for field f . The displacement term averages the normalized error equally over active samples:

$$\mathcal{L}_u = \frac{1}{|\mathcal{A}_u|} \sum_{n \in \mathcal{A}_u} \frac{\sum_j M_{n,j,u} [(\hat{y}_{n,j,u} - y_{n,j,u})/s_{n,u}]^2}{\sum_j M_{n,j,u}} \quad (24)$$

For a transition threshold $\delta > 0$, the Huber penalty is [27]

$$H_\delta(e) = \begin{cases} \frac{1}{2}e^2, & |e| \leq \delta, \\ \delta(|e| - \frac{\delta}{2}), & |e| > \delta \end{cases} \quad (25)$$

The scalar-field losses apply the Huber penalty with $\delta = 1$ to the RMS-normalized errors and aggregate the contributions over all active nodes in the batch:

$$\mathcal{L}_f = \frac{\sum_{n,j} M_{n,j,f} H_1((\hat{y}_{n,j,f} - y_{n,j,f})/s_{n,f})}{\max(\sum_{n,j} M_{n,j,f}, 1)}, f \in \{\varepsilon_{eq}, \sigma_{vm}\} \quad (26)$$

Equal weight is assigned to each sample in the displacement loss and to each included node in the scalar losses within a batch. Samples are not resampled according to RMS amplitude.

2.3.3. Peak and amplitude penalties

High-response regions are emphasized using a bounded spatial weight. For $f \in \{u, \varepsilon_{eq}, \sigma_{vm}\}$, the FEM nodal amplitude is $a_{n,i,u} = \|\mathbf{u}_{n,i}\|_2$ for displacement and $a_{n,i,f} = |y_{n,i,f}|$ for a scalar field. Predicted amplitudes are denoted by $\hat{a}_{n,i,f}$. Using the sample scale $s_{n,f}$ from Section 2.3.2,

$$w_{n,i,f} = \begin{cases} 1 + (a_{n,i,f}/s_{n,f})^2, & a_{n,i,f} \leq 3s_{n,f}, \\ 10, & a_{n,i,f} > 3s_{n,f} \end{cases} \quad (27)$$

The set $\mathcal{J}_{n,f}$ contains retained node-component pairs for displacement and retained nodes for either scalar field. The weighted relative squared-error penalty is

$$\mathcal{L}_{peak,f} = \frac{1}{|\mathcal{A}_f|} \sum_{n \in \mathcal{A}_f} \frac{\sum_{j \in \mathcal{J}_{n,f}} w_{n,j,f} (\hat{y}_{n,j,f} - y_{n,j,f})^2}{\max\left(\sum_{j \in \mathcal{J}_{n,f}} w_{n,j,f} y_{n,j,f}^2, \epsilon_r^2\right)} \quad (28)$$

For displacement, the sample scale is the RMS over Cartesian components, whereas the nodal amplitude is a vector norm. Consequently, the weight-saturation threshold of three component-RMS units is $\sqrt{3}$ times the field-level vector RMS. The active set contains samples with nonempty retained sets. $\epsilon_r = 10^{-6}$, and all displacement components at a node share the same weight. The amplitude term uses adaptive pooling to at most 16 cells per axis. For each nonempty cell, the index set $\mathcal{I}_{n,k,f}$ contains nodes with at least one retained field component, giving

$$\bar{a}_{n,k,f} = \frac{\sum_{i \in \mathcal{I}_{n,k,f}} a_{n,i,f}}{|\mathcal{I}_{n,k,f}|}, \bar{\hat{a}}_{n,k,f} = \frac{\sum_{i \in \mathcal{I}_{n,k,f}} \hat{a}_{n,i,f}}{|\mathcal{I}_{n,k,f}|} \quad (29)$$

Let $q_{n,f}$ and $\hat{q}_{n,f}$ be the 95th percentiles of the FEM and predicted cell averages, using linear interpolation between ordered values. Their logarithmic ratio is penalized by

$$\mathcal{L}_{amp,f} = \frac{1}{|\mathcal{A}_f|} \sum_{n \in \mathcal{A}_f} H_{0.1} \left[\log \left(\frac{\hat{q}_{n,f} + \epsilon_r}{q_{n,f} + \epsilon_r} \right) \right] \quad (30)$$

The quantile term constrains amplitude rather than peak location. Empty samples are omitted, and a penalty is zero if none remains. Peak and amplitude penalties are evaluated every fourth and eighth mini-batch, respectively. For zero-based mini-batch index t ,

$$\kappa_p(t) = \begin{cases} 4, & t = 4k, \\ 0, & \text{otherwise,} \end{cases} \quad \kappa_a(t) = \begin{cases} 8, & t = 8k, \\ 0, & \text{otherwise,} \end{cases} \quad k = 0, 1, 2, \dots \quad (31)$$

The multipliers preserve the mean coefficients over a complete evaluation cycle. The resulting gradients nevertheless differ from those obtained by evaluating every term in every mini-batch. The training objective is

$$\begin{aligned} \mathcal{L}(\sqcup) = & \mathcal{L}_u + \mathcal{L}_{BC} + 0.5\mathcal{L}_{\varepsilon_{eq}} + 0.5\mathcal{L}_{\sigma_{vm}} \\ & + 0.25\kappa_p(t) \sum_f \mathcal{L}_{peak,f} + 0.05\kappa_a(t) \sum_f \mathcal{L}_{amp,f} \end{aligned} \quad (32)$$

Validation and test losses evaluate every term with $\kappa_p = \kappa_a = 1$. At inference, all five channels are predicted directly and prescribed zero-displacement constraints are imposed on the displacement output.

3. Results and Discussion

3.1. Experimental design

Low-resolution models were evaluated against two attention-free volumetric CNNs, and four configurations of backbone width and output heads were compared. The low-resolution cases retain the governing equations, geometries, loads, supports, and field definitions. The reference model uses 16 base channels, bottleneck global attention, and one local-window attention block shared by the three fields. The alternatives reduce the base width to eight channels, use CNN scalar-output heads, or use separate local-window attention blocks for each field. These four configurations share 61,523 training and 7,690 validation samples.

Each of these four configurations was trained for 5,000 epochs using AdamW, a peak learning rate of 8×10^{-4} , and weight decay of 10^{-4} , with biases and normalization parameters exempt from decay. Five warm-up epochs preceded cosine decay to 10^{-6} without restarts. Samples were grouped by padded shape, without random cropping or spatial augmentation. All four configurations were evaluated after epoch 5,000 on the same test cases using the common settings in Table 1. Throughout the results, benchmark coordinates and displacements are reported as x/L_0 and u/L_0 , strain is dimensionless, and stress is reported as σ/E_0 . Because $L_0 = 1$ m and $E_0 = 1$ Pa, nondimensionalization leaves the stored numerical values unchanged.

Displacement error uses all valid nodes except load-marker and support-marker nodes. Strain and stress errors additionally require nodal density $\rho \geq 0.5$ to evaluate the material region. Padding is excluded throughout. Let $\mathbf{y}_{n,f}$ and $\hat{\mathbf{y}}_{n,f}$ contain the retained entries $\mathcal{J}_{n,f}$ of field f in case n . The case-level relative RMSE and its test-set mean are

Table 1. Common training settings for the low-resolution architecture comparisons.

Parameter	Setting
Training epochs	5,000
Optimizer and weight decay	AdamW, 10^{-4}
Learning-rate schedule	8×10^{-4} peak, 5-epoch warm-up, cosine to 10^{-6}
Displacement response interval	$10^{-3} \leq \text{valid-node displacement RMS} \leq 1$
Target scaling	Per-sample loss normalization
Evaluation epoch	5,000
Precision	FP32
Batching	Grouped by padded spatial shape
Spatial augmentation	Disabled for architecture ablations

$$r_{n,f} = \frac{\|\hat{\mathbf{y}}_{n,f} - \mathbf{y}_{n,f}\|_2}{\max\left(\|\mathbf{y}_{n,f}\|_2, 10^{-12}\sqrt{|\mathcal{I}_{n,f}|}\right)}, \bar{r}_f = \frac{1}{N_{test}} \sum_{n=1}^{N_{test}} r_{n,f} \quad (33)$$

Displacement entries are the three Cartesian components, not displacement magnitudes. The common entry-count factor cancels between RMSE and target RMS in Eq. (33). The main test comparison averages case-level relative errors equally. Medians and upper quantiles describe their distribution. Mean absolute error is calculated within each case and then averaged across cases. Uncertainty in the test-set estimates was quantified using a source-group bootstrap [28]. Each of 4,000 replicates samples the 5,120 source groups with replacement and retains all available variants of each selected group. The same resampling is used for every architecture, permitting paired differences. The 2.5th and 97.5th percentiles provide 95% intervals. These intervals quantify test-sampling uncertainty for the trained networks.

Percentage mean absolute error (PMAE) and percentage peak absolute error (PPAE) are calculated per case [7]:

$$PMAE_{n,f} = \frac{\sum_{i \in I_{n,f}} |y_{n,f,i} - \hat{y}_{n,f,i}|}{N_{n,f} [\max_{i \in I_{n,f}} y_{n,f,i} - \min_{i \in I_{n,f}} y_{n,f,i}]} \times 100\% \quad (33a)$$

$$PPAE_{n,f} = \frac{|\max_{i \in I_{n,f}} y_{n,f,i} - \max_{i \in I_{n,f}} \hat{y}_{n,f,i}|}{\max_{i \in I_{n,f}} y_{n,f,i}} \times 100\% \quad (33b)$$

where $N_{n,f} = |I_{n,f}|$ is the number of retained entries for field f in case n . Cases with zero reference range for PMAE or zero reference maximum for PPAE are undefined for the corresponding metric. The same field-specific masks used for relative RMSE are applied. Displacement PMAE and PPAE use nodal displacement magnitude, whereas displacement relative RMSE retains the three signed Cartesian components.

3.2. Benchmarking against volumetric CNN architectures

Predictive accuracy was compared with that of attention-free volumetric CNNs on the common low-resolution test set. A residual 3D U-Net, an encoder-only 3D ResNet dense regressor, and the shared-attention and task-specific-attention models were evaluated on 7,688 cases after 5,000 training epochs. The field masks, case-level relative RMSE, and source-group bootstrap defined in Section 3.1 were used for all four architectures (Table 2).

The residual 3D U-Net gives lower mean relative RMSE than the encoder-only 3D ResNet: 0.2210 versus 0.2535 for displacement, 0.1837 versus 0.4647 for equivalent strain, and 0.1762 versus 0.4633 for von Mises stress. Compared with the residual 3D U-Net, the shared-attention model decreases these errors by 13.6%, 25.6%, and 24.1%, respectively. The task-specific-attention model gives displacement error 0.1937 and the lowest strain and stress errors in this comparison, 0.1337 and 0.1305. The shared- and task-specific-attention models contain 7.807 and 7.831 million parameters, compared with 6.019 million for the residual 3D U-Net. The comparison is between complete architectures, with different parameter counts.

3.3. Effect of backbone width and output heads

The effects of backbone width and output-head design were examined using the same training and evaluation protocol (Table 3 and Fig. 4).

The 16-channel shared-attention model gives mean relative RMSEs of 0.1910, 0.1367, and 0.1337 for displacement, equivalent strain, and von Mises stress. Reducing the base width to

Table 2. Comparison with attention-free volumetric CNN baselines on the common low-resolution test set after 5,000 training epochs. Brackets give 95% source-group bootstrap intervals. The 3D ResNet is a ResNet-18-style encoder-only dense regressor with trilinear output upsampling.

Architecture	Parameters (M)	Displacement rRMSE ↓	Equivalent strain rRMSE ↓	von Mises stress rRMSE ↓
Encoder-only 3D ResNet	2.076	0.2535 [0.2484, 0.2589]	0.4647 [0.4626, 0.4670]	0.4633 [0.4612, 0.4656]
Residual 3D U-Net	6.019	0.2210 [0.2145, 0.2277]	0.1837 [0.1810, 0.1866]	0.1762 [0.1698, 0.1871]
Shared-attention 3DSolidFormer	7.807	0.1910 [0.1849, 0.1973]	0.1367 [0.1345, 0.1391]	0.1337 [0.1316, 0.1360]
Task-specific-attention 3DSolidFormer	7.831	0.1937 [0.1878, 0.1999]	0.1337 [0.1319, 0.1357]	0.1305 [0.1287, 0.1324]

Table 3. Mean case-level relative RMSE for the low-resolution ablation study with 95% source-group bootstrap intervals. Lower errors are better. Intervals reflect resampling of held-out source groups.

Architecture	Parameters (M)	u	ϵ_{eq}	σ_{vm}
Shared attention, 16 channels	7.807	0.1910 [0.1849, 0.1973]	0.1367 [0.1345, 0.1391]	0.1337 [0.1316, 0.1360]
Shared attention, 8 channels	1.996	0.2466 [0.2384, 0.2555]	0.1915 [0.1880, 0.1954]	0.1883 [0.1849, 0.1922]
CNN output heads, 16 channels	7.868	0.2249 [0.2180, 0.2323]	0.1834 [0.1813, 0.1857]	0.1720 [0.1698, 0.1741]
Task-specific attention, 16 channels	7.831	0.1937 [0.1878, 0.1999]	0.1337 [0.1319, 0.1357]	0.1305 [0.1287, 0.1324]

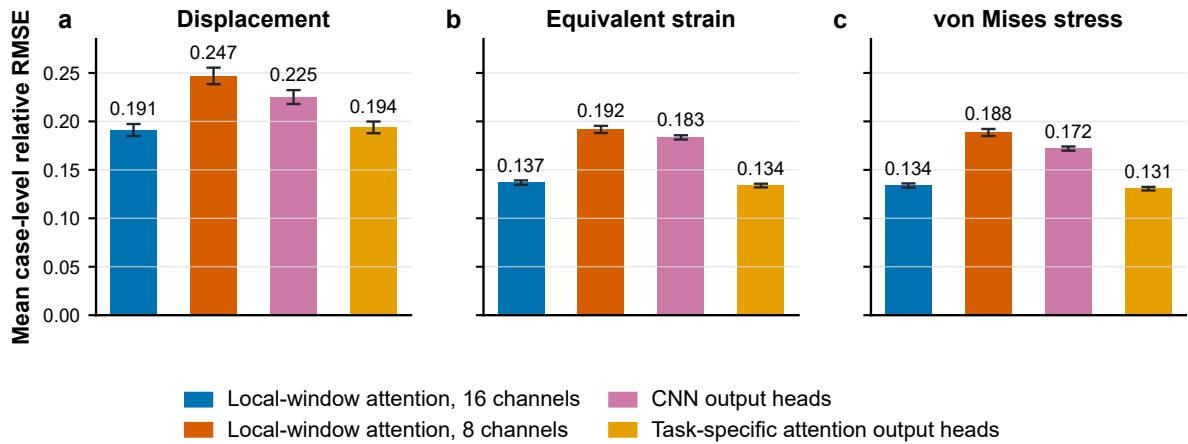


Fig. 4. Mean case-level relative RMSE in the low-resolution ablation study for (a) displacement, (b) equivalent strain, and (c) von Mises stress. Error bars show 95% percentile intervals from 4,000 paired source-group bootstrap replicates. Displacement errors use valid evaluation nodes. Scalar errors additionally require nodal density of at least 0.5.

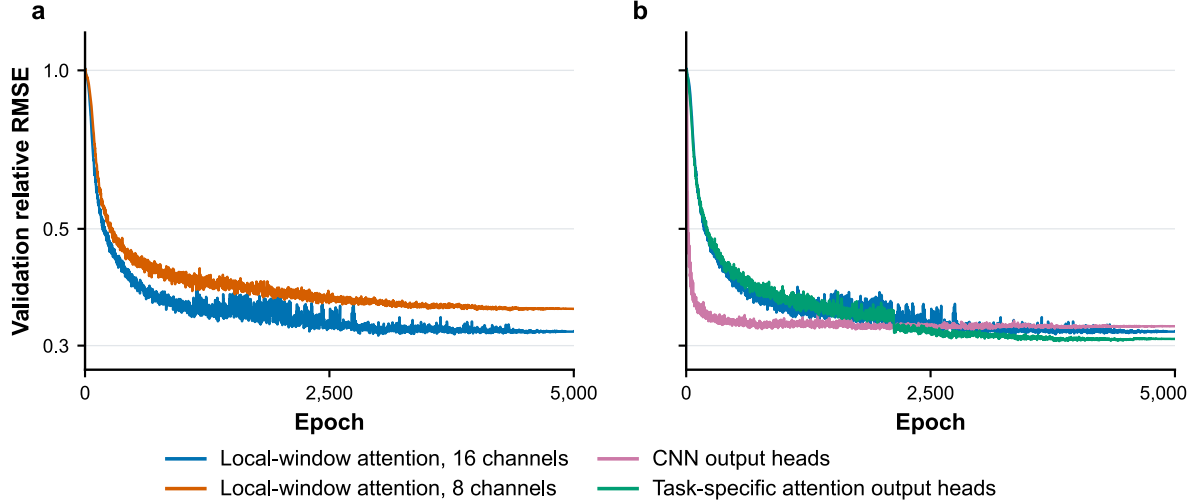


Fig. 5. Displacement validation histories for the low-resolution ablation study. (a) Shared local-window attention models with 16 and eight base channels. (b) Shared-attention, CNN, and task-specific-attention output heads. Curves cover 5,000 epochs. Validation relative RMSE is pooled over field entries, whereas Table 3 reports the mean of case-level test errors.

eight channels decreases the parameter count from 7.807 to 1.996 million and increases the corresponding errors to 0.2466, 0.1915, and 0.1883. Relative to the 16-channel model, displacement, strain, and stress errors increase by approximately 29.1%, 40.1%, and 40.9%, respectively.

The task-specific-attention model gives mean relative RMSEs of 0.1937, 0.1337, and 0.1305 with 7.831 million parameters. Relative to the 16-channel shared-attention model, the displacement-error difference is 0.0027, with a paired 95% interval of $[-0.0014, 0.0066]$. The corresponding strain and stress differences are -0.0030 $[-0.0042, -0.0019]$ and -0.0032 $[-0.0045, -0.0021]$. The displacement interval includes zero, whereas both scalar-field intervals exclude zero.

Independent CNN output heads increase mean displacement, strain, and stress errors to 0.2249, 0.1834, and 0.1720. Their parameter count, 7.868 million, is close to that of the 16-channel shared-attention model. The original inputs and a stop-gradient displacement prediction are supplied to the CNN branches. The comparison therefore reflects the complete output-head design, including its inputs and gradient paths.

Displacement validation histories are compared in Fig. 5 for the shared-attention models with 8 and 16 base channels and for the three output-head designs. The 16-channel model has lower displacement validation relative RMSE than the reduced-width model over most of training. The final validation errors of the three output-head configurations are similar, and validation error decreases more slowly in later epochs. The plotted validation metric is the pooled RMSE divided by the pooled FEM RMS, rather than the mean of case-level relative errors. Tables 2 and 3 report the fixed-epoch test comparison after 5,000 epochs.

3.4. Dependence on response amplitude and geometry

A small proportion of cases have substantially larger errors, particularly for displacement (Table 4). For the 16-channel shared-attention model, the median displacement relative RMSE is 0.1166 and its 95th percentile is 0.5807. Equivalent strain and stress have medians of 0.1209 and 0.1188, with 95th percentiles of 0.2498 and 0.2396. All finite test cases are included.

Grouping cases by displacement RMS reveals larger relative errors at small response amplitudes. In the RMS interval 10^{-3} to 10^{-2} , the median displacement relative RMSE for the 16-

Table 4. Error distribution of the 16-channel shared-attention model over 7,688 low-resolution test cases. MAE uses the same normalized units as the figures: displacement divided by L_0 , dimensionless equivalent strain, and stress divided by E_0 . An rRMSE above one exceeds the error of a zero prediction on the same nonzero FEM field.

Field	Mean MAE	Median rRMSE	95th percentile rRMSE	Maximum rRMSE
Displacement	0.011877	0.1166	0.5807	5.840
Equivalent strain	0.00030521	0.1209	0.2498	3.106
von Mises stress	0.00033309	0.1188	0.2396	3.044

channel shared-attention model is 0.4488 across 126 cases, compared with 0.0969 across 2,763 cases in 0.1 to 0.5. The intermediate intervals 0.01 to 0.05 and 0.05 to 0.1 contain 1,796 and 2,992 cases, respectively. Only ten cases lie in 0.5 to 1. A single case falls below 10^{-3} under the evaluation mask because dataset screening used all valid nodes, whereas evaluation excludes loaded and constrained nodes.

Topology-optimized cases are also more difficult on average. For the 16-channel shared-attention model, mean displacement relative RMSE is 0.1630 on fully solid cases and 0.2374 on topology-optimized cases. The corresponding strain errors are 0.1289 and 0.1495, and stress errors are 0.1288 and 0.1417. Geometry complexity and response amplitude vary together in these groups, preventing their effects from being separated in this comparison. Relative errors remain larger at low response amplitudes despite per-sample loss normalization.

3.5. Spatial accuracy of the predicted fields

Agreement in response amplitude is not sufficient to establish agreement in field shape. For each case, let $\alpha_n = \|\hat{\mathbf{u}}_n\|_2 / \|\mathbf{u}_n\|_2$ and let c_n be the cosine between the retained predicted and FEM displacement vectors. When the FEM norm and the product of the predicted and FEM norms exceed their numerical floors,

$$r_{n,u}^2 = 1 + \alpha_n^2 - 2\alpha_n c_n \quad (34)$$

The cosine calculation uses a floor of 10^{-24} on the product of the two norms. For a zero prediction and a nonzero FEM response above the denominator floor in Eq. (33), $\alpha_n = 0$, $c_n = 0$, $r_{n,u} = 1$. The recorded zero cosine is a numerical convention and does not define a direction for the zero vector.

The 16-channel shared-attention model has a mean displacement RMS ratio of 0.9874 and mean cosine similarity of 0.9723. The corresponding medians are 0.9983 and 0.9955. These statistics indicate close average agreement in displacement amplitude and direction, although large errors remain in individual cases. Equation (34) relates these quantities for each case, not their separate test-set averages.

FEM and predicted equivalent strain and normalized von Mises stress are compared on orthogonal sections through a ribbed topology-optimized structure from the low-resolution test set (Fig. 6). The same sections are used for the shared-attention, CNN, and task-specific-attention output heads. The main strain and stress concentrations are reproduced by all three designs. Each model's checkpoint with the lowest validation displacement relative RMSE is used for the spatial comparisons, whereas Table 3 uses the fixed epoch-5,000 checkpoints. To show lower-amplitude variations alongside localized maxima, field colors are mapped as

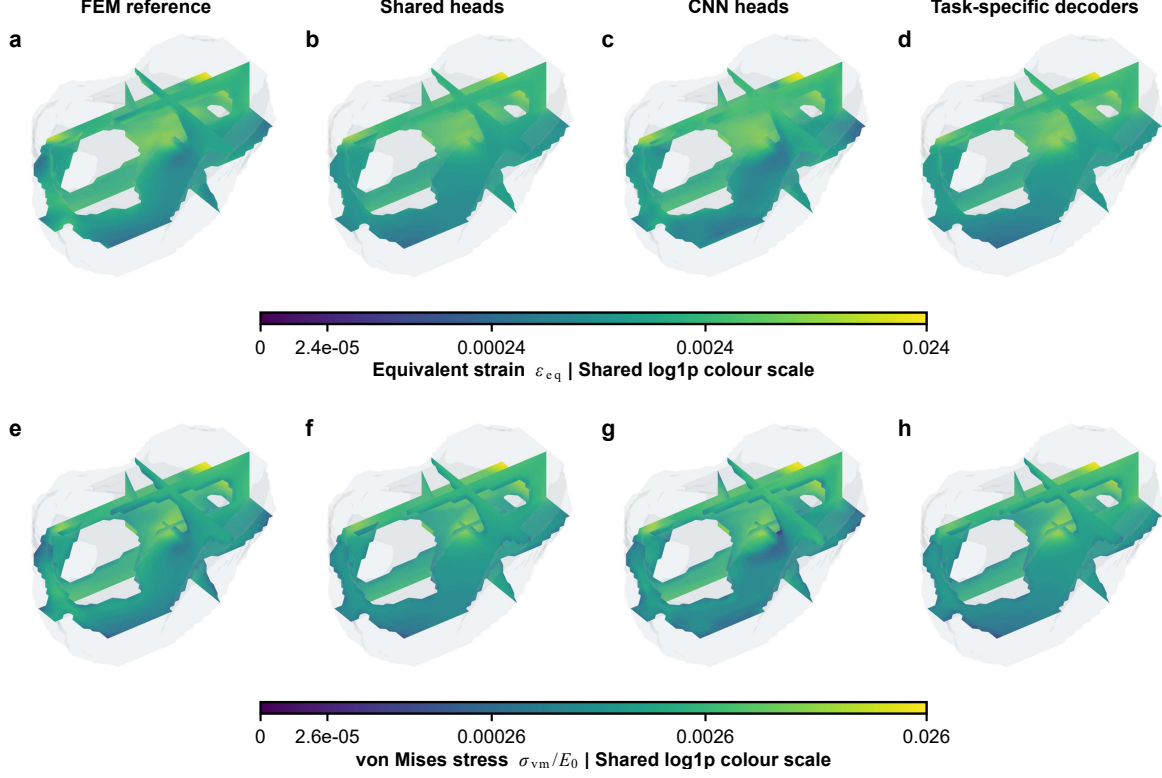


Fig. 6. Interior strain and stress comparison on ribbed source 37544. Equivalent strain is shown in panels (a)-(d) and normalized von Mises stress in panels (e)-(h). Columns show FEM, local-window self-attention heads, CNN heads, and task-specific local-window self-attention heads. All columns use identical geometry, orthogonal sections ($z = 8$, $y = 16$, $x = 21$), camera, and field-specific color scales. Colors use the normalized logarithmic mapping in Eq. (35), with $q_0 = 10^{-3}q_{\max}$ determined separately for each field from the largest displayed section value across FEM and all three models. The scales are $q_0 = 2.403 \times 10^{-5}$ for ε_{eq} and $q_0 = 2.553 \times 10^{-5}$ for σ_{vm}/E_0 . Colorbar ticks show the original dimensionless field values, not the transformed values.

$$c(q) = \frac{\log_{10}(1 + q/q_0)}{\log_{10}(1 + q_{\max}/q_0)}, \quad \text{with } q_0 = 10^{-3}q_{\max} \quad (35)$$

where $q_{\max}$ is the shared upper color limit. This mapping is equivalent to Astropy's LogStretch with $a = 1000$ [29]. Colorbar ticks show original dimensionless values.

For a second topology-optimized structure from the low-resolution test set, normalized von Mises stress from FEM and the same three output-head designs is compared on two orthogonal sections in Fig. 7, with signed prediction errors on a third section. The main high-stress regions are reproduced by all three designs. Larger localized underprediction is visible with the CNN heads in the signed-error section, despite the similar overall field distributions.

Structural similarity (SSIM) [30] complements amplitude-sensitive error by comparing local means, variances, and covariance. For SSIM calculation at center i , the Gaussian neighbor weights w_{ij} are renormalized over valid material nodes M_j ,

$$\alpha_{ij} = \frac{w_{ij}M_j}{\sum_{k \in W_i} w_{ik}M_k}, \quad \mu_{t,i} = \sum_{j \in W_i} \alpha_{ij}t_j \quad (36)$$

Prediction means, population variances, and covariance use the same weights. The 11×11 Gaussian window has a standard deviation of 1.5 nodes. At image boundaries, windows are

truncated and the remaining weights are renormalized. With FEM target t and prediction p , the local SSIM is

$$S_i = \frac{(2\mu_{t,i}\mu_{p,i} + C_1)(2\sigma_{tp,i} + C_2)}{(\mu_{t,i}^2 + \mu_{p,i}^2 + C_1)(\sigma_{t,i}^2 + \sigma_{p,i}^2 + C_2)} \quad (37)$$

where $C_1 = (0.01L)^2$ and $C_2 = (0.03L)^2$ [30], with L equal to the FEM maximum minus minimum on the evaluated section and common to all compared models for each field. SSIM is the arithmetic mean over valid centers. The quantity $1 - \text{SSIM}$ measures dissimilarity, not a percentage accuracy. A center requires support of at least three nodes. Displacement-magnitude SSIM cannot detect vector-direction errors and therefore complements rather than replaces component-wise displacement evaluation.

Local field variation is evaluated using relative scalar L2 error $\|p - t\|_2 / \|t\|_2$ and an adjacent-node gradient error on the same material support. If E contains adjacent valid node pairs within the plane and $\delta q_{ij} = (q_j - q_i)/h$, then

$$E_{\nabla} = \frac{\|\delta p - \delta t\|_{2,E}}{\|\delta t\|_{2,E}} \quad (38)$$

Pairs crossing a void or an excluded node are omitted. Equation (38) requires at least one eligible pair and a nonzero reference-gradient norm. The relative gradient error is undefined if either condition is absent. Spatial variation in the plotted scalar field is measured by this quantity.

A common planar section through a third topology-optimized structure from the low-resolution test set is used in Fig. 8 to compare FEM and predicted equivalent strain and normalized von Mises stress. Field maps and signed errors are compared for the shared-attention, CNN, and task-specific-attention output heads. Among the three model predictions, the highest SSIM values are obtained with task-specific attention for both fields. Larger localized signed errors are visible with the CNN heads, particularly near the high-response region.

Figure 9 compares FEM and predicted strain and stress along the material-only profile marked in Fig. 8. It also shows adjacent-node differences along the profile and cumulative distributions over the entire selected plane. The profile spans 31 contiguous material nodes and 30 adjacent pairs. All three models reproduce the main variations in equivalent strain and von Mises stress along the profile. The CNN heads show the largest departures from FEM, while task-specific attention generally remains closest. The cumulative distributions are calculated using all 901 material nodes on the selected plane. The spatial comparisons show that all three output-head designs reproduce the main field patterns, with larger localized departures for the CNN heads.

3.6. High-resolution prediction

Task-specific attention was selected for high-resolution training following the low-resolution comparisons, in which the lowest equivalent-strain and von Mises stress errors were obtained with comparable displacement accuracy. All parameters of the trained low-resolution model were fine-tuned for 55 epochs using high-resolution inputs and FEM targets. The trained network returns displacement, equivalent strain, and von Mises stress on the high-resolution mesh in a single forward pass. Mesh spacing and boundary-condition patches follow Section 2.1.3. Fine-tuning uses AdamW with weight decay 10^{-4} , a learning rate of 5×10^{-5} .

On the 7,488-case high-resolution test set, median relative RMSEs are 0.1172 for displacement components, 0.1475 for equivalent strain, and 0.1448 for von Mises stress. The corresponding 95th percentiles are 0.5800, 0.2770, and 0.2680. The lower median relative RMSE of 0.0775 for displacement magnitude reflects the omission of directional differences from that

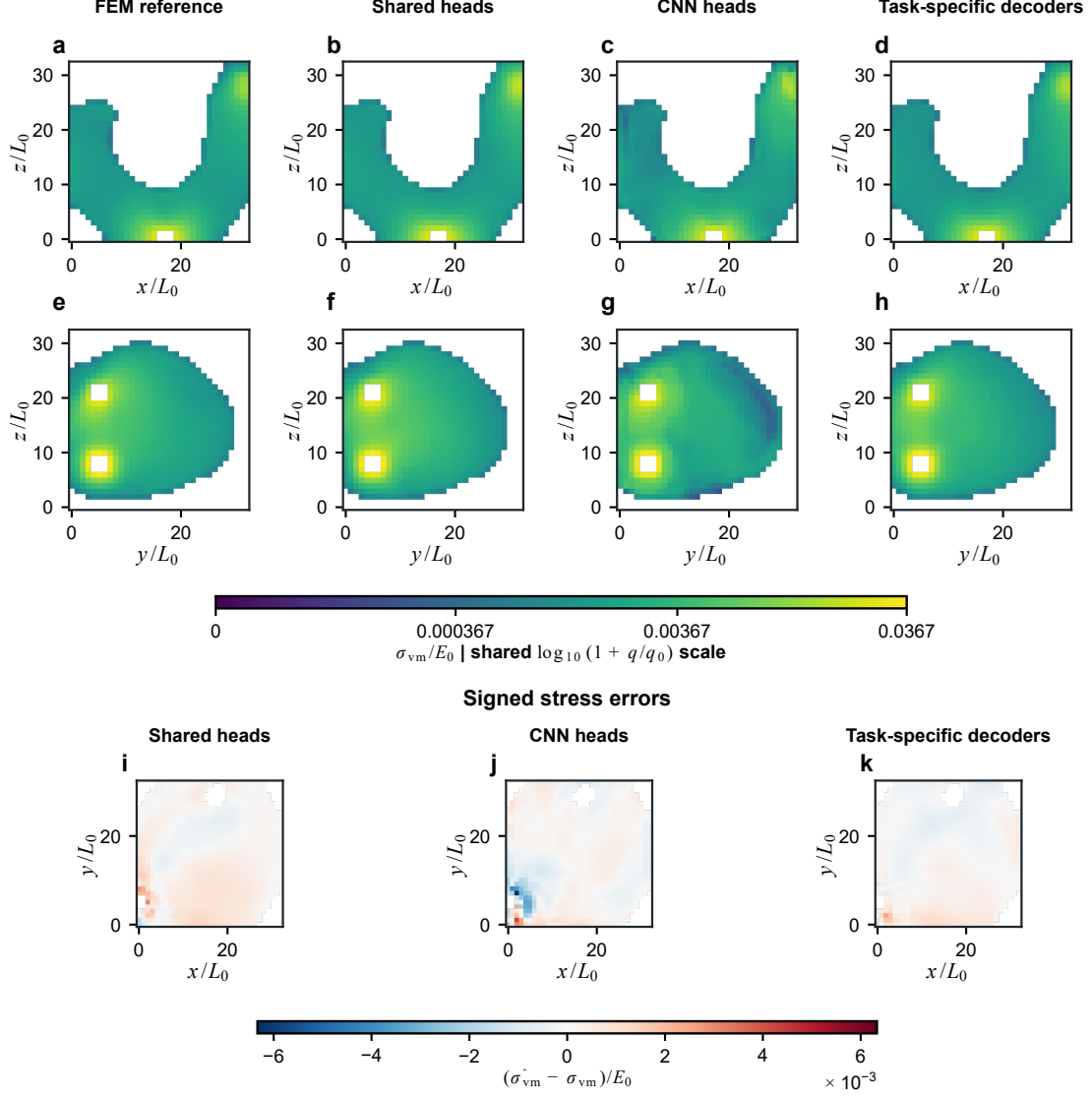


Fig. 7. Orthogonal stress sections and signed errors for source 59272. Panels (a)-(h) show normalized stress on two orthogonal sections, and panels (i)-(k) show signed stress errors on a third orthogonal section. Columns compare FEM, local-window self-attention heads, CNN heads, and task-specific local-window self-attention heads where applicable. Field panels (a)-(h) use the normalized logarithmic mapping in Eq. (35), with $q = \sigma_{vm}/E_0$ and $q_0 = 10^{-3}q_{\max}$, giving $q_0 = 3.672 \times 10^{-5}$. Here, $q_{\max}$ is the largest displayed stress value over both sections and all four columns. Colorbar ticks show original normalized stress values. Error panels (i)-(k) use a separate linear scale symmetric about zero. Red denotes overprediction and blue underprediction. Evaluated material nodes have nodal density $\rho \geq 0.5$ and exclude padding and nodes carrying applied loads or displacement constraints.

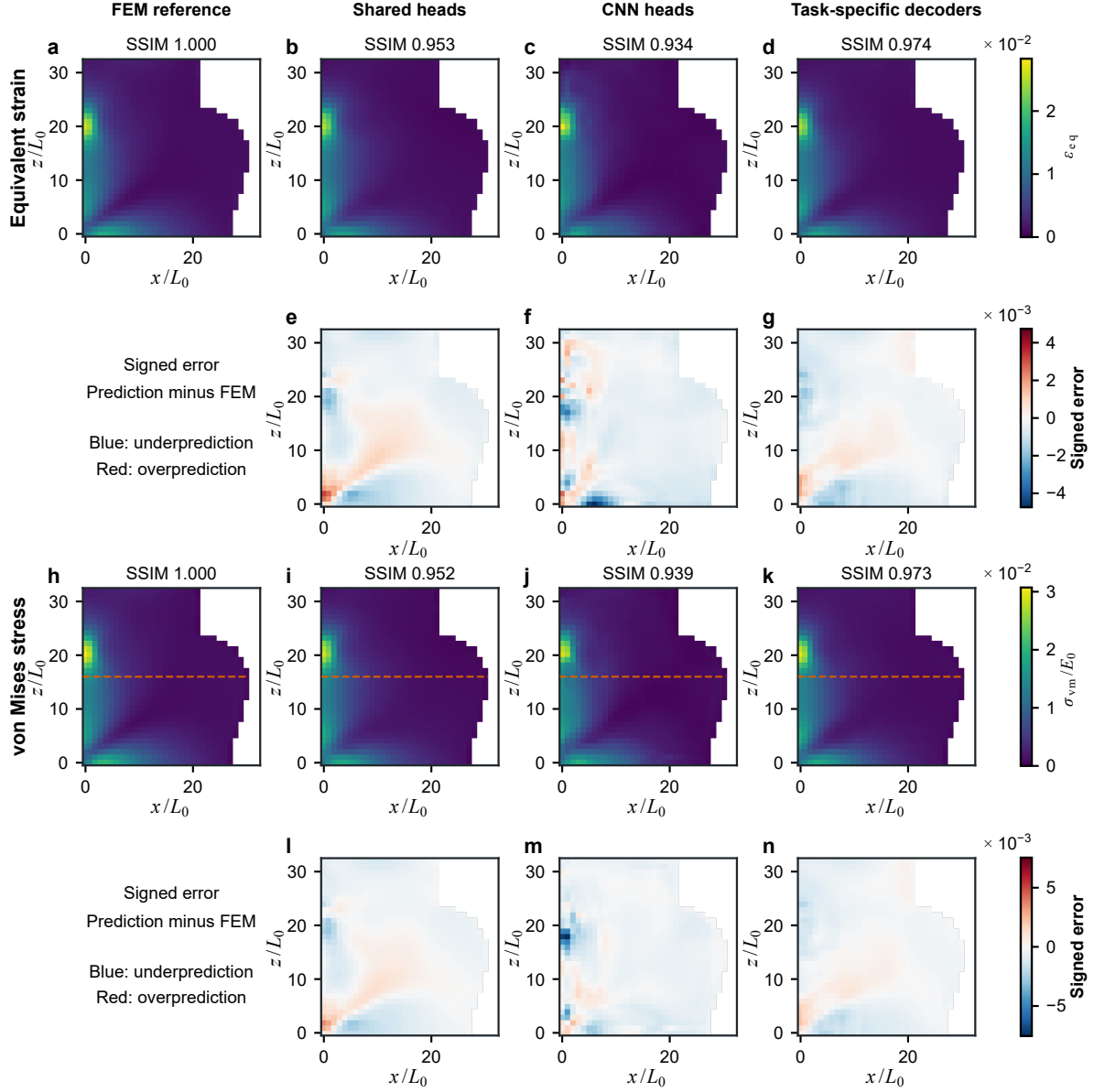


Fig. 8. Strain and stress fields and signed errors for topology-optimized source 38361 on the selected y-index 8 plane. Panels (a) to (g) show equivalent strain and its signed errors. Panels (h) to (n) show normalized von Mises stress and its signed errors. Columns compare FEM, local-window self-attention heads, CNN heads, and task-specific local-window self-attention heads. Field rows share one scale and error rows share one symmetric scale. The orange guide marks the z-index 16 profile evaluated in Fig. 9.

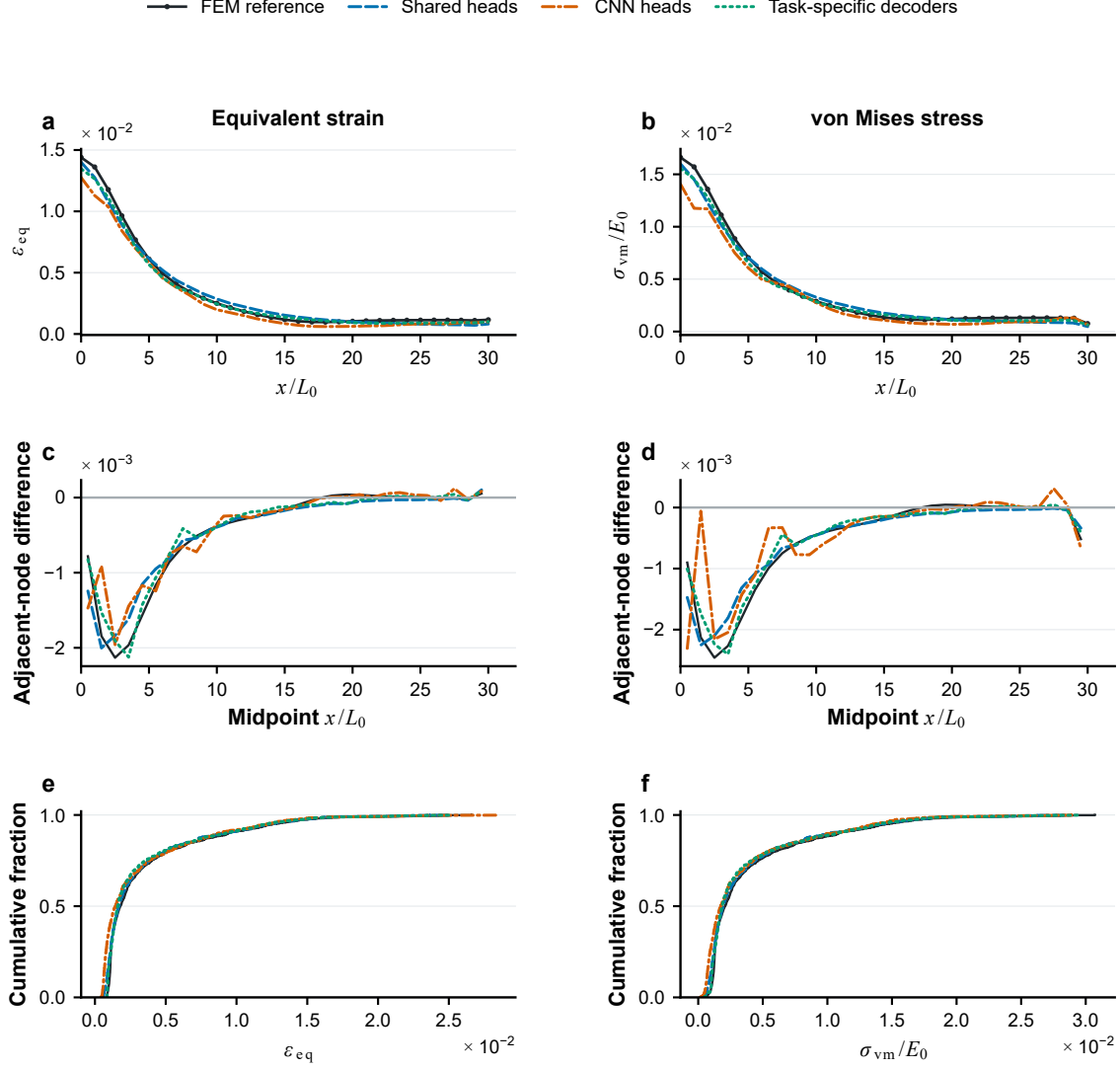


Fig. 9. Strain and stress profiles and distributions for source 38361. Equivalent strain is shown in panels (a), (c), and (e), and normalized von Mises stress in panels (b), (d), and (f). The first row follows x on the selected y -index 8 plane at z -index 16, the second shows adjacent-node differences, and the third shows empirical cumulative distributions over all 901 material nodes. Identical nodes are used for FEM and all models.

metric. Median PMAE values are 1.329% for displacement magnitude, 0.153% for equivalent strain, and 0.151% for von Mises stress. Median PPAE values are 7.588%, 7.695%, and 7.527%, respectively (Table 5). Mean PMAE values are 2.62% for displacement magnitude, 0.204% for equivalent strain, and 0.198% for von Mises stress.

Direct prediction gives lower strain and stress errors than recovery from predicted displacement. Median PMAE decreases from 0.655% to 0.153% for equivalent strain and from 0.653% to 0.151% for von Mises stress, reductions of 76.6% and 76.9%, respectively. Median PPAE decreases from 27.40% to 7.70% for strain and from 27.41% to 7.53% for stress. Median relative RMSEs decrease from 0.5701 to 0.1475 for strain and from 0.5703 to 0.1448 for stress. Applying the same recovery procedure to FEM displacement gives median relative RMSEs of 2.28×10^{-7} for both scalar fields across the 7,488 test cases. Agreement with the stored scalar targets supports the recovery implementation. The larger errors from predicted displacement are consistent with amplification of local displacement errors by spatial differentiation.

Table 5. High-resolution error distribution over all 7,488 held-out test cases. Medians are reported because the case-level error distributions are right-skewed. The 95th percentile characterizes upper-tail performance. PMAE and PPAE are medians of case-level percentages. Displacement relative RMSE uses the three signed Cartesian components, whereas displacement PMAE and PPAE use nodal displacement magnitude. Scalar-field metrics use the material-region evaluation mask.

Field	Median rRMSE	95th percentile rRMSE	Median PMAE (%)	Median PPAE (%)
Displacement	0.1172	0.5800	1.329	7.588
Equivalent strain	0.1475	0.2770	0.153	7.695
von Mises stress	0.1448	0.2680	0.151	7.527

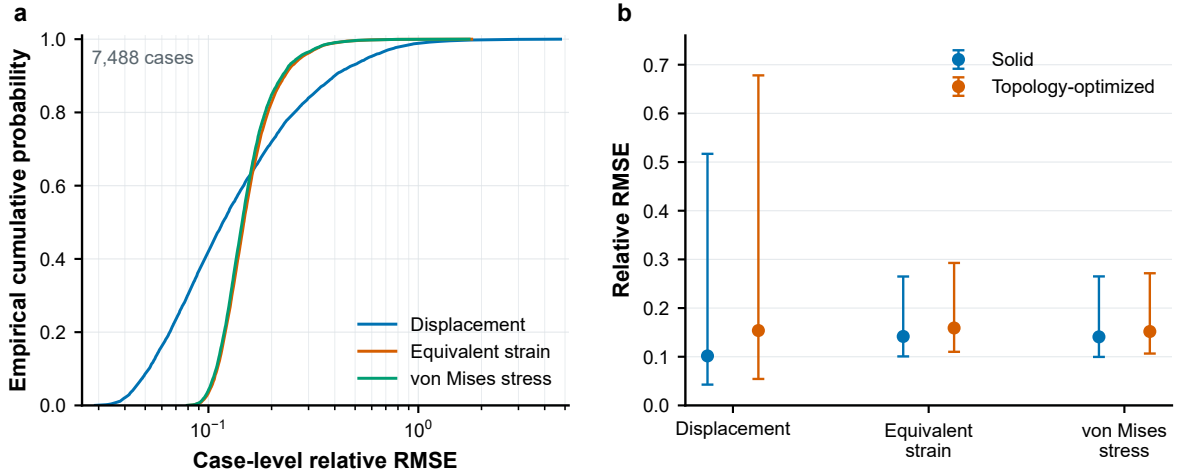


Fig. 10. High-resolution error distributions after fine-tuning. (a) Empirical cumulative distributions of case-level relative RMSE for component-wise displacement, equivalent strain, and von Mises stress across 7,488 test cases. (b) Geometry-stratified medians and 5th to 95th percentile ranges.

Inference and recovery were timed separately on one Tesla V100 GPU using 20 evenly spaced test cases, two warm-up iterations, and five timed repetitions per case. BF16 inference was used for timing, while fine-tuning and field-error evaluation used FP32. The median evaluation time for the five output channels was 0.412 s per case. Recovery from predicted displacement required an additional 0.075 s, with a combined median of 0.487 s. Data loading and host-to-device transfer were excluded.

The high-resolution test set contains 2,709 topology-optimized and 4,779 fully solid cases. Mean relative RMSEs for topology-optimized cases are 0.2339, 0.1755, and 0.1675 for displacement, strain, and stress, compared with 0.1657, 0.1566, and 0.1558 for fully solid cases. Figure 10 shows the empirical error distributions and geometry-stratified medians and 5th to 95th percentile ranges.

Figure 11 compares surface fields of displacement magnitude, equivalent strain, and normalized von Mises stress from FEM and 3DSolidFormer for four topology-optimized structures from the high-resolution test set. Strain and stress are concentrated near loading patches, while lower values extend over most of the material surface.

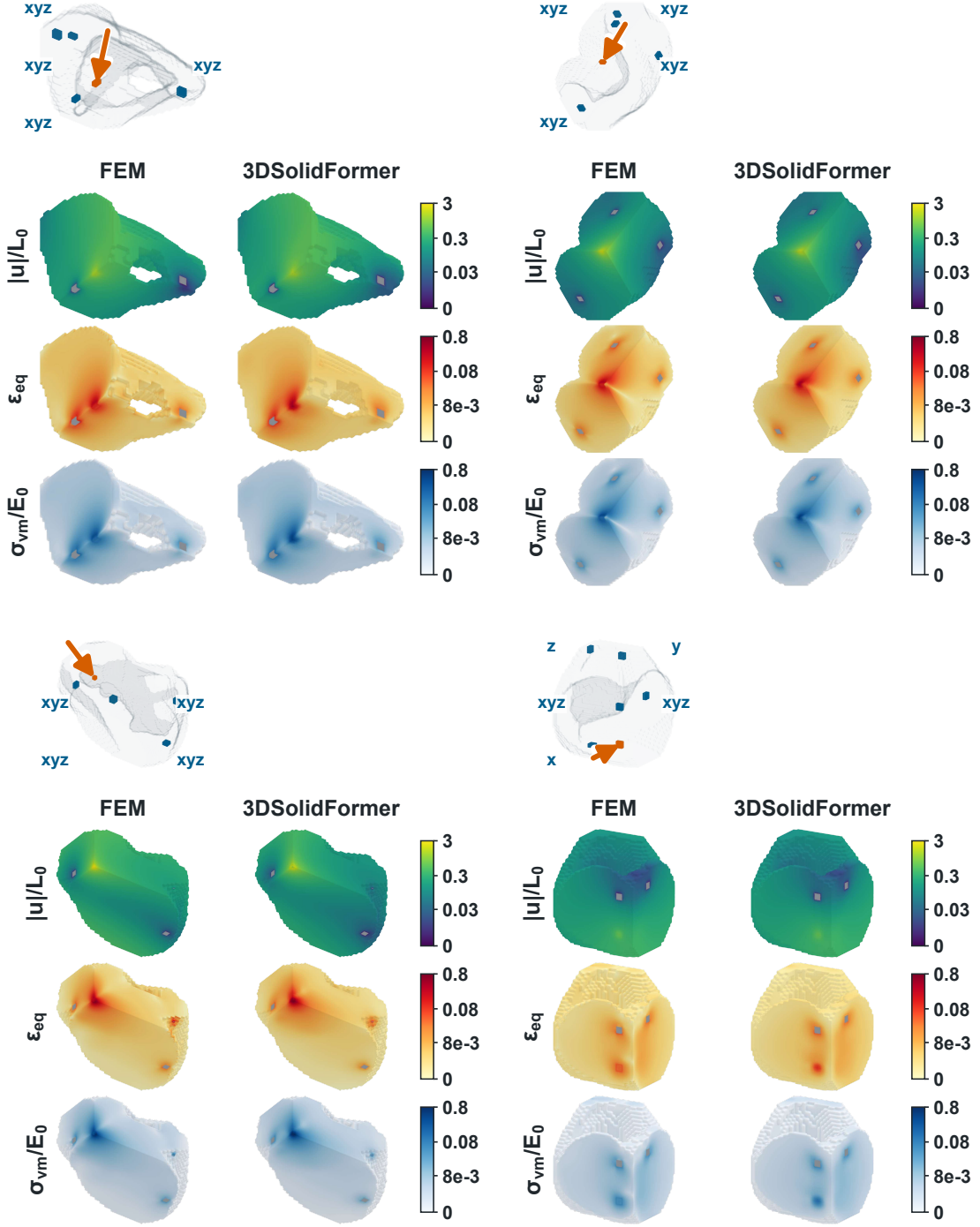


Fig. 11. Native high-resolution FEM and 3DSolidFormer surface fields for four topology-optimized test geometries. Rows show dimensionless displacement magnitude, equivalent strain, and normalized von Mises stress. Each field shares one scale across all cases and both columns. Colors use the normalized logarithmic mapping in Eq. (35), with $q_0 = 10^{-3}q_{\max}$, as in Figs. 6 and 7. Here, $q_{\max}$ is the upper color limit, giving $q_0 = 3 \times 10^{-3}$ for displacement and $q_0 = 8 \times 10^{-4}$ for strain and stress. Ticks show original dimensionless values. Orange arrows denote loads. Blue patches and x, y, z labels indicate constrained components.

4. Conclusions

Joint prediction of displacement, equivalent strain, and von Mises stress was demonstrated on three-dimensional domains using 3DSolidFormer. The 3DSolidFormer architecture combines global bottleneck attention with field-specific local-window attention, while prescribed zero-displacement constraints are satisfied by construction. After high-resolution fine-tuning, median case-level relative RMSEs were 0.1172, 0.1475, and 0.1448 across 7,488 test cases. Median PMAE values were 1.33% for displacement magnitude, 0.153% for equivalent strain, and 0.151% for von Mises stress. Corresponding median PPAE values were 7.59%, 7.70%, and 7.53%. Relative to recovery from predicted displacement, direct scalar-field prediction reduced median PMAE by 76.6% for strain and 76.9% for stress. These gains were obtained while returning all three fields in a single forward pass and enforcing prescribed zero displacements.

The accompanying dataset contains 75,114 finite-element cases with material distributions, loads, supports, and nodal response fields for fully solid and topology-optimized structures. Each computational domain contains 2.05 to 3.07 million voxel elements, including solid and void regions. Source identities are aligned across low- and high-resolution partitions, permitting consistent benchmarking and investigation of learning across mesh resolutions without regenerating the finite-element training data.

Lower field errors were obtained with the attention-based architectures than with the evaluated attention-free volumetric CNNs. Within 3DSolidFormer, task-specific attention gave modest improvements in strain and stress relative to a shared local-window block, with comparable displacement performance within the reported test-sampling uncertainty. Reducing the backbone width removed 74.4% of the parameters but increased errors in all three fields. These comparisons establish the performance of the tested architectures without isolating the effect of attention from other design differences.

The feasibility of joint full-field prediction across millions of voxels was demonstrated within the evaluated range of small-strain linear-elastic problems. In future work, the approach will be extended to multicomponent systems, and the prediction model will be integrated into the developed high-resolution multicriteria topology optimization frameworks. Mechanical consistency and design-relevant quantities will be assessed within the optimization loops.

References

- [1] A.I.J. Forrester, A. Sóbester, A.J. Keane, Engineering Design via Surrogate Modelling: A Practical Guide, Engineering Design via Surrogate Modelling: A Practical Guide (2008) 1–223. <https://doi.org/10.1002/9780470770801>.
- [2] P. Benner, S. Gugercin, K. Willcox, A survey of projection-based model reduction methods for parametric dynamical systems, SIAM Rev. 57 (2015) 483–531. <https://doi.org/10.1137/130932715>.
- [3] F. Chinesta, P. Ladeveze, E. Cueto, A Short Review on Model Order Reduction Based on Proper Generalized Decomposition, Archives of Computational Methods in Engineering 2011 18:4 18 (2011) 395–404. <https://doi.org/10.1007/S11831-011-9064-7>.
- [4] M. Raissi, P. Perdikaris, G.E. Karniadakis, Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations, J. Comput. Phys. 378 (2019) 686–707. <https://doi.org/10.1016/j.jcp.2018.10.045>.
- [5] S. Wang, Y. Teng, P. Perdikaris, Understanding and mitigating gradient flow pathologies in physics-informed neural networks, SIAM Journal on Scientific Computing 43 (2021) 3055–3081. <https://doi.org/10.1137/20M1318043>.
- [6] Z. Nie, H. Jiang, L.B. Kara, Stress field prediction in cantilevered structures using convolutional neural networks, J. Comput. Inf. Sci. Eng. 20 (2020). <https://doi.org/10.1115/1.4044097/955168>.

- [7] H. Jiang, Z. Nie, R. Yeo, A.B. Farimani, L.B. Kara, StressGAN: A generative deep learning model for two-dimensional stress distribution prediction, *Journal of Applied Mechanics, Transactions ASME* 88 (2021). <https://doi.org/10.1115/1.4049805/1096635>.
- [8] T. Pfaff, M. Fortunato, A. Sanchez-Gonzalez, P.W. Battaglia, LEARNING MESH-BASED SIMULATION WITH GRAPH NETWORKS, *ICLR 2021 - 9th International Conference on Learning Representations* (2021).
- [9] L. Lu, P. Jin, G. Pang, Z. Zhang, G.E. Karniadakis, Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators, *Nat. Mach. Intell.* 3 (2021) 218–229. <https://doi.org/10.1038/S42256-021-00302-5>.
- [10] J. He, S. Koric, S. Kushwaha, J. Park, D. Abueidda, I. Jasiuk, Novel DeepONet architecture to predict stresses in elastoplastic structures with variable complex geometries and loads, *Comput. Methods Appl. Mech. Eng.* 415 (2023) 116277. <https://doi.org/10.1016/J.CMA.2023.116277>.
- [11] S. Cao, Choose a Transformer: Fourier or Galerkin, *Adv. Neural Inf. Process. Syst.* 34 (2021) 24924–24940. <https://github.com/scaomath/galerkin-transformer> (accessed September 6, 2026).
- [12] Z. Li, K. Meidani, A.B. Farimani, Transformer for Partial Differential Equations’ Operator Learning, *Transactions on Machine Learning Research* 2023 (2022). <https://arxiv.org/pdf/2205.13671> (accessed September 6, 2026).
- [13] E. Lejeune, Mechanical MNIST: A benchmark dataset for mechanical metamodels, *Extreme Mech. Lett.* 36 (2020) 100659. <https://doi.org/10.1016/J.EML.2020.100659>.
- [14] E. Whalen, A. Beyene, C. Mueller, SimJEB: Simulated Jet Engine Bracket Dataset, *Computer Graphics Forum* 40 (2021) 9–17. <https://doi.org/10.1111/CGF.14353>; JOURNAL: JOURNAL:14678659; W GROUP: STRING: PUBLICATION.
- [15] S. Hong, Y. Kwon, D. Shin, J. Park, N. Kang, DeepJEB: 3D Deep Learning-Based Synthetic Jet Engine Bracket Dataset, *Journal of Mechanical Design* 147 (2025). <https://doi.org/10.1115/1.4067089/1208546>.
- [16] A.H. Nobari, L. Regenwetter, F. Ahmed, Towards Domain-Adaptive, Resolution-Free 3D Topology Optimization With Neural Implicit Fields, *Proceedings of the ASME Design Engineering Technical Conference* 3A-2024 (2024). <https://doi.org/10.1115/DETC2024-142135>.
- [17] T.J.R.. Hughes, *The finite element method: linear static and dynamic finite element analysis*, Prentice-Hall, 1987.
- [18] M. MacKlin, Warp: A High-performance Python Framework for GPU Simulation and Graphics, (2022). <https://github.com/nvidia/warp> (accessed September 7, 2026).
- [19] M.R. Hestenes, E. Stiefel, Methods of conjugate gradients for solving linear systems, *J. Res. Natl. Bur. Stand.* (1934). 49 (1952) 409. <https://doi.org/10.6028/JRES.049.044>.
- [20] Y. Saad, *Iterative Methods for Sparse Linear Systems*, Society for Industrial and Applied Mathematics, 2003. <https://doi.org/10.1137/1.9780898718003>.
- [21] Ö. Çiçek, A. Abdulkadir, S.S. Lienkamp, T. Brox, O. Ronneberger, 3D U-net: Learning dense volumetric segmentation from sparse annotation, *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 9901 LNCS (2016) 424–432. https://doi.org/10.1007/978-3-319-46723-8_49/TABLES/3.
- [22] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE Computer Society, 2016: pp. 770–778. <https://doi.org/10.1109/CVPR.2016.90>.
- [23] Y. Wu, K. He, Group Normalization, in: *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.
- [24] D. Hendrycks, K. Gimpel, Gaussian Error Linear Units (GELUs), (2023). <http://arxiv.org/abs/1606.08415> (accessed August 12, 2026).

- [25] A.D. Jagtap, K. Kawaguchi, G.E. Karniadakis, Locally adaptive activation functions with slope recovery term for deep and physics-informed neural networks, *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 476 (2020). <https://doi.org/10.1098/rspa.2020.0334>.
- [26] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention Is All You Need, *Adv. Neural Inf. Process. Syst.* 2017-December (2017) 5999–6009. <https://doi.org/10.48550/arxiv.1706.03762>.
- [27] P.J. Huber, Robust Estimation of a Location Parameter, (1992) 492–518. https://doi.org/10.1007/978-1-4612-4380-9_35.
- [28] B. Efron, Bootstrap Methods: Another Look at the Jackknife, *Ann. Statist.* 7 (1979) 1–26. <https://doi.org/10.1214/AOS/1176344552>.
- [29] LogStretch — Astropy v8.0.2, (n.d.). <https://docs.astropy.org/en/stable/api/astropy.visualization.LogStretch.html> (accessed October 21, 2026).
- [30] Z. Wang, A.C. Bovik, H.R. Sheikh, E.P. Simoncelli, Image quality assessment: From error visibility to structural similarity, *IEEE Transactions on Image Processing* 13 (2004) 600–612. <https://doi.org/10.1109/TIP.2003.819861>.