
GEOTRANSOLVER: LEARNING PHYSICS ON IRREGULAR DOMAINS USING MULTI-SCALE GEOMETRY AWARE PHYSICS ATTENTION TRANSFORMER

Corey Adams¹, Rishikesh Ranade ^{*}, Ram Cherukuri¹, and Sanjay Choudhry¹

¹NVIDIA Corporation, San Jose, California, USA

ABSTRACT

We present GeoTransolver, a Multiscale Geometry-Aware Physics Attention Transformer for CAE that replaces standard attention with GALE, coupling physics-aware self-attention on learned state slices with cross-attention to a shared geometry/global/boundary-condition context computed from multi-scale ball queries (inspired by DoMINO) and reused in every block. Implemented and released in NVIDIA PhysicsNeMo, GeoTransolver persistently projects geometry, global and boundary condition parameters into physical state spaces to anchor latent computations to domain structure and operating regimes. We benchmark GeoTransolver on DrivAerML, Luminary SHIFT-SUV, and Luminary SHIFT-Wing, comparing against Domino, Transolver (as released in PhysicsNeMo), and literature-reported AB-UPT, and evaluate drag/lift R^2 and Relative L_1 errors for field variables. GeoTransolver delivers better accuracy, improved robustness to geometry/regime shifts, and favorable data efficiency; we include ablations on DrivAerML and qualitative results such as contour plots and design trends for the best GeoTransolver models. By unifying multiscale geometry-aware context with physics-based attention in a scalable transformer, GeoTransolver advances operator learning for high-fidelity surrogate modeling across complex, irregular domains and non-linear physical regimes.

Keywords Transformers · Multiscale Geometry Aware Models · CAE Surrogate Modeling

1 Introduction

Computer-aided engineering (CAE) increasingly depends on AI-based surrogates to accelerate design exploration to reduce reliance on costly high-fidelity simulation [1, 2, 3, 4]. However, building reliable AI models for physics remains a challenge. Real-world geometries are heterogeneous and irregular with multiscale features; highly non-linear physical phenomena arising from local effects can produce long-range dependencies which span the entire computational domain; physical regimes and boundary conditions must be encoded and represented efficiently; and models must respect physical constraints (e.g., incompressibility with $\nabla u = 0$) without sacrificing performance and scalability. Data is often scarce or biased by solver settings, domain shifts are common across shape families and physical regimes, and deep operator learners can accumulate error layer-by-layer, particularly in stiff regimes [5]. These factors have limited the stability, robustness, generalization, and trustworthiness of existing AI based approaches.

This paper introduces GeoTransolver, a Multiscale Geometry-Aware Physics Attention Transformer for CAE applications. GeoTransolver extends the Transolver [6] backbone by replacing standard attention with GALE (Geometry-Aware Latent Embeddings) attention, which unifies physics-aware self-attention on learned state slices with cross-attention to geometry and global context embeddings. Inspired from Domino’s multi-scale ball query formulations [7], GeoTransolver learns global geometry encodings and local latent encodings that capture neighborhoods at multiple radii, preserving fine-grained near-boundary behavior and far-field interactions. Crucially, geometry and global features are projected into physical state spaces and injected as context in every transformer block, ensuring persistent conditioning and alignment between evolving latent states and the underlying domain.

^{*}Corresponding author: rranade@nvidia.com

GALE directly targets core challenges in AI physics modeling. By structuring self-attention around physics-aware slices, GeoTransolver encourages interactions that reflect operator couplings (e.g., pressure–velocity or field–material). Multi-scale ball queries enforce locality where needed while maintaining access to global signals, balancing efficiency with nonlocal reasoning. Continuous geometry-context projection at depth mitigates representation drift and improves stability, while providing a natural interface for constraint-aware training and regularization. Together, these design choices enhance accuracy, robustness to geometric and regime shifts, and scalability on large, irregular discretizations.

GeoTransolver is implemented using the NVIDIA PhysicsNeMo framework [8], which provides geometry data pipelines, multi-physics conditioning utilities, and scalable training/inference infrastructure. PhysicsNeMo enables efficient integration of multi-scale neighborhood construction, global context handling, and distributed training necessary for high-resolution CAE workloads.

We benchmark GeoTransolver against state-of-the-art architectures - Domino [7], AB-UPT [9], and Transolver [6] on three widely used CAE datasets: DrivAerML [10], Luminary SHIFT-SUV [11], and Luminary SHIFT-Wing [12]. The benchmarking is carried out using the framework developed in PhysicsNeMo [13]. These tasks span aerodynamic flows over parameterized vehicle and wing geometries across diverse operating regimes. We evaluate field-level reconstruction (e.g., velocity, pressure and wall shear) and integral quantities (e.g., drag and lift coefficients), along with generalization to unseen geometries and design trends. Across these benchmarks, GeoTransolver demonstrates better accuracy, improved data efficiency, and strong robustness, underscoring the value of geometry-aware, globally conditioned attention.

Our key contributions are:

1. GeoTransolver, a transformer architecture with GALE attention that pairs physics-aware self-attention on learned state slices with cross-attention to multi-scale geometric neighborhoods and global context.
2. A geometry-context projection strategy (inspired from Transolver [6]) that maps geometry and global features into physical state spaces and injects them into every block, enabling persistent conditioning and reducing layer-wise drift.
3. An adaptation of multi-scale ball queries (inspired from Domino [7]) that balances local fidelity and global coupling on irregular meshes and point clouds, improving efficiency and stability.
4. A comprehensive benchmark on DrivAerML, SHIFT-SUV, and SHIFT-Wing against Domino, AB-UPT, and Transolver, demonstrating gains in accuracy.

2 Background and Related Work

AI-based surrogates for CAE span operator learning, mesh/geometry-aware representations, multi-scale architectures, and physics-informed training. Below we survey the most relevant lines of work and position GeoTransolver relative to them.

2.1 Neural operators

Neural operators learn mappings between function spaces, enabling resolution-agnostic surrogates for PDEs [14]. DeepONet [15] pioneered branch–trunk architectures for continuous operators, while Fourier Neural Operator (FNO) [16, 17] introduced spectral convolution to capture nonlocal interactions efficiently. Variants extend FNO with learnable bases, adaptive grids, or hybrid spectral–spatial blocks, and graph-based neural operators (GNO) (citations) target irregular discretizations via message passing on meshes [18, 19]. Transformer-based operators have shown to improve long-range coupling and scalability due to the self- and cross- attention capability in many scientific disciplines [20]. Transformer neural operators [21, 22, 23, 24, 25, 6] apply global attention over latent fields, sometimes with sparsification or hierarchical tokens to manage cost. Other efforts integrate attention with spectral operators or U-Net hierarchies to balance locality and global mixing.

GeoTransolver builds on this trajectory but replaces standard attention with GALE, structuring self-attention around physics-aware state slices while using cross-attention to geometry and global context. This differs from purely field-centric attention by persistently anchoring computations to domain and regime signals at every block.

2.2 Geometry and mesh-aware encoders

MeshGraphNet and related GNNs [26, 27, 28, 29] popularized message passing on simulation meshes, capturing local interactions via edge neighborhoods and respecting mesh topology. Extensions incorporate learnable stencils,

anisotropic kernels, and multigrid coarsening to handle large geometries [30, 31]. Point-cloud networks [32, 33, 34] introduced multi-scale representations and local neighborhood aggregation for unstructured geometry, with hierarchical encoders that preserve fine detail while summarizing global shape. Domino [7] recently demonstrated that multi-scale ball queries paired with global shape descriptors can significantly improve generalization across shape families in CAE settings. AB-UPT [9] explores unified physics transformers with geometry-grounded tokens and attention that mixes local neighborhoods with global descriptors.

GeoTransolver adopts multi-scale ball query formulations to learn both local latent encodings and global geometry encodings, but goes further by projecting geometry and global features into physical state spaces and reusing them as context in all transformer blocks, not only at input or at a fixed hierarchy level.

2.3 Multi-scale locality and global coupling

U-Net-style hierarchies, graph coarsening, and multigrid networks remain standard tools to reconcile boundary-layer fidelity with far-field interactions [35, 36, 37]. Spectral methods [16, 38] offer efficient global mixing but can struggle on irregular domains without careful remeshing. Hierarchical transformers and sparse attention schemes target efficiency by limiting attention to neighborhoods or learned clustering, reintroducing global tokens when needed.

GeoTransolver’s GALE attention uses multi-scale neighborhoods for efficiency while preserving a dedicated channel for global context via cross-attention, ensuring nonlocal couplings are available throughout depth without relying solely on global tokens at the input.

2.4 Physics-aware conditioning and constraints

Physics-informed neural networks (PINNs) [39] and Physics-Informed Neural Operators (PINO) [40] add residual or Sobolev losses to enforce PDE constraints and boundary conditions. Equivariant architectures encode symmetry (e.g., SE(3) equivariance), while divergence-free or curl-free parameterizations target invariants such as incompressibility. Conditional operator learning introduces regime parameters (e.g., Reynolds/Mach, materials) and boundary metadata as inputs or conditioning vectors; however, many models inject context only once, which can lead to representation drift across layers.

GeoTransolver’s geometry-context projection supplies global and boundary information as aligned physical-state context at every block, providing persistent conditioning that improves stability and extrapolation without mandating hard constraints. Although not used in this work, the design is compatible and flexible with PINN/PINO-style losses and regularizers for future implementation.

2.5 AI modeling for CAE

Overall, in the context of CAE the literature points to three converging needs: geometry awareness on irregular domains, multi-scale locality with nonlocal coupling, and sustained conditioning on global context and constraints. GeoTransolver’s combination of GALE attention, multi-scale ball queries, and geometry-context projection operationalizes these requirements within a scalable transformer backbone, contributing a complementary path alongside neural operators, GNNs, and hierarchical transformers.

GeoTransolver adopts crucial knowledge from all the state-of-the-art architectures, but also introduces unique components that enable accurate and efficient learning. With respect to DoMINO and AB-UPT, GeoTransolver adopts multi-scale ball queries but introduces GALE, which explicitly pairs physics-aware self-attention on state slices with cross-attention to geometry and global context at every layer. This persistent conditioning is designed to reduce drift and improve robustness under geometry/regime shifts. Related to Transolver, GeoTransolver replaces standard global attention with GALE and adds geometry-context projection, enhancing geometric grounding and context retention across the forward pass. Finally, GeoTransolver targets irregular geometries without remeshing and emphasizes multi-scale geometric neighborhoods, while maintaining transformer flexibility for nonlocal coupling. It can be trained with operator-style supervision and augmented with physics-informed losses like models such as FNO, GNO and DeepONet.

3 GeoTransolver

GeoTransolver operates on one or more local input streams (‘slices’) and a shared geometry/global context. Each GALE block performs slice-wise self-attention and cross-attention to the precomputed context, with an adaptive gate

that blends the two. Multi-scale ball-query features are computed once before the first block and appended to local inputs when enabled. The shared context is computed once and reused at every depth.

The inputs to GeoTransolver are defined as follows:

- Local inputs (slices): for slice m with N_m tokens, positions and features are:

$$X_m = \{(x_{m,i}, f_{m,i})\}_{i=1}^{N_m}, \quad x_{m,i} \in \mathbb{R}^3, \quad f_{m,i} \in \mathbb{R}^{d_x} \quad (1)$$

- Geometry:

$$\mathcal{G} = \{(g_j, \gamma_j)\}_{j=1}^{M_g}, \quad g_j \in \mathbb{R}^3, \quad \gamma_j \in \mathbb{R}^{d_g} \quad (2)$$

- Global parameters:

$$p \in \mathbb{R}^{d_p} \quad (3)$$

3.1 Geometry Aware Latent Embeddings (GALE)

The physics attention mechanism pioneered by Transolver offers a high quality mechanism to deploy attention-based learning on high resolution scientific data, and especially on unstructured data: using learnable projections from state features to latent space ‘Physical States’ provides a highly flexible mechanism for input points to learn relationships between each other, without the quadratic computational and memory behavior of full attention [6]. However, since the inputs to each physics attention layer are the outputs of the previous layer only, there is no ‘recall’ ability for fine grained attending to encode geometrical features. To resolve this, we introduce a modification to the physics attention layer that incorporates global and geometrical information, encoded directly to the ‘Physical State’ latent space, referred to here as ‘Geometry Aware Latent Embeddings’ or GALE.

The key deviations for a GALE layer occur after the latent state self attention, as seen in Figure 1. The learned projections for a given layer are used in a cross-attention mechanism with the ‘context’ embedding, learned directly from the input points and geometries. This cross-attention is then mixed into the output slices via a learnable parameter, allowing the model to adapt it’s focus per layer towards a global context or a self-attention context, as needed. After the mixing, the latent points are ‘de-sliced’ (as in Transolver [6]) back to the input dimensionality.

3.2 Learning Geometrical Latent Embeddings

To enable the model to better incorporate geometrical information, GeoTransolver follows a two stage process to prepare inputs for processing. The goals of this feature extraction are first to create a geometry-informed latent space, by projecting the global and geometrical information via a PhysicsAttention ‘slice’ operation. No ‘deslicing’ is done, here: instead this output state is left as a geometrical and global state latent vector, which is used in cross attention in each GALE layer.

Additionally, during the creation of the geometrical context, GeoTransolver performs a set of radius-bounded ball-query samplings similar to DoMINO architecture [7]. For each point in the input space, up to ‘ k ’ (configurable) points are sampled from the geometry and processed via an MLP. These are appended to the input space at each point, to enhance the initial geometric information, prior to the first GALE layer. Furthermore, for each point in the geometry, GeoTransolver extracts up to ‘ k ’ points from the input space, processes them with an MLP, and slices them to a latent vector that is concatenated with the global geometry latent vector.

The multi-scale radii and neighborhoods are defined as:

$$\mathcal{S} = \{(r_s, k_s)\}_{s=1}^S \quad (4)$$

The geometry-to-input ball queries augment local inputs.

$$\mathcal{N}_{m,i,s}^{\text{geom}} = \{ j : \|x_{m,i} - g_j\| \leq r_s \}_{\leq k_s}, \quad h_{m,i,s}^{\text{BQ}} = \psi_{m,s}\left(\{[\gamma_j, g_j - x_{m,i}]\}_{j \in \mathcal{N}_{m,i,s}^{\text{geom}}}^{\text{geom}}\right) \quad (5)$$

$$U_{m,i} = [h_{m,i,1}^{\text{BQ}}, \dots, h_{m,i,S}^{\text{BQ}}], \quad U_m = \{U_{m,i}\}_{i=1}^{N_m} \quad (6)$$

On the otherhand, input-to-geometry ball queries summarize context across slices

$$\mathcal{N}_{j,s}^{\text{inp}} = \{ (n, i) : \|g_j - x_{n,i}\| \leq r_s \}_{\leq k_s}, \quad h_{j,s}^{\text{inp}} = \varphi_s\left(\{[f_{n,i}, x_{n,i} - g_j]\}_{(n,i) \in \mathcal{N}_{j,s}^{\text{inp}}}^{\text{inp}}\right) \quad (7)$$

$$E_s = \text{Pool}_j h_{j,s}^{\text{inp}}, \quad c_{\text{geom}} = \text{Pool}_j \rho(\gamma_j), \quad C = [p, c_{\text{geom}}, E_1, \dots, E_S] \in \mathbb{R}^{d_c} \quad (8)$$

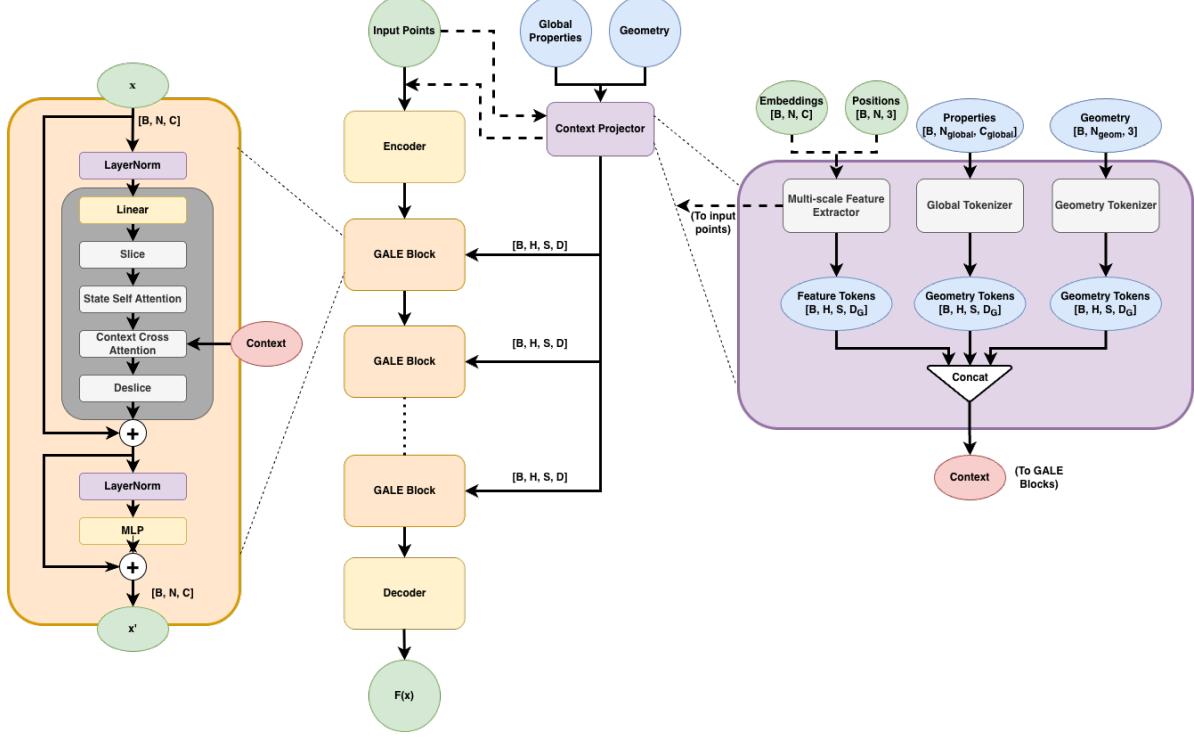


Figure 1: The GeoTransolver model architecture. The main contributions of GeoTransolver are twofold: first, a geometry- and global-state-aware context projector, to augment input points and encode a sample context. Second, the addition of cross attention in the PhysicsAttention Layer, known here as a Geometry Aware Latent Embedding layer or GALE layer.

Here, $\psi_{m,s}$, φ_s and ρ and Pool is a permutation-invariant reducer (mean/max/attention). The shared context C is reused at all depths. Multi-scale ball queries follow the spirit of DoMINO’s neighborhoods, enabling local and global geometry capture across radii and k values [7].

Both radius-bounded ball-query operations are multi-scale such that different MLPs are constructed for a different radii and ‘ k ’ values. Smaller radii enable learning local information while larger radii enable global information propagation. This multi-scale operation enables capturing local and global geometry information in the geometry-aware context.

Local preprocessing and augmentation Per-slice features are projected to latent space; optional multi-scale augmentation is appended once (before the first GALE block).

$$H_m^{(0)} = P_m(\{f_{m,i}\}_i), \quad \tilde{H}_m^{(0)} = \text{Concat}(H_m^{(0)}, Q_m(U_m)) \quad (9)$$

In each GALE block, $l = 1, \dots, L$ and slice m the following operations are carried out:

- Slice-wise self-attention:

$$\text{SA}_m^{(\ell)} = \text{Attn}(\tilde{H}_m^{(\ell-1)} W_Q^{(\ell,m)}, \tilde{H}_m^{(\ell-1)} W_K^{(\ell,m)}, \tilde{H}_m^{(\ell-1)} W_V^{(\ell,m)}) \quad (10)$$

- Cross-attention to shared context:

$$\text{CA}_m^{(\ell)} = \text{Attn}(\tilde{H}_m^{(\ell-1)} W_{Q,c}^{(\ell,m)}, C W_{K,c}^{(\ell,m)}, C W_{V,c}^{(\ell,m)}) \quad (11)$$

- Adaptive gate and mixing:

$$\alpha_m^{(\ell)} = \sigma(\eta^{(\ell,m)}(\text{Pool}(\text{SA}_m^{(\ell)}), \text{Pool}(C))) \in (0, 1), \quad \hat{H}_m^{(\ell)} = (1 - \alpha_m^{(\ell)}) \text{SA}_m^{(\ell)} + \alpha_m^{(\ell)} \text{CA}_m^{(\ell)} \quad (12)$$

- Feed-forward update with residual:

$$\tilde{H}_m^{(\ell)} = \hat{H}_m^{(\ell)} + \text{MLP}^{(\ell,m)}(\hat{H}_m^{(\ell)}) \quad (13)$$

Here, $W_{Q/K/V}^{(l,m)}$ and $W_{Q/K/V,c}^{(l,m)}$ are learned projections, $\eta^{(l,m)}$ is a small gating network, σ is sigmoid, and $\text{Attn}(Q, K, V) = \text{softmax}(QK^\top / \sqrt{d_k}) V$.

After mixing, Transolver-style ‘de-slicing’ back to field dimensionality can be implemented via per-slice heads (consistent with the current forward path). Outputs Per-slice normalization and output heads produce final predictions:

$$Y_m = \text{MLP}_{\text{out}}^{(m)} \left(\text{LN}^{(m)} \left(\tilde{H}_m^{(L)} \right) \right) \quad (14)$$

The goal of GALE is to enable geometry-aware context throughout the model. The latent vector, which is used as cross-attention input in all subsequent GeoTransolver layers, is built not just from the points sampled from a geometry mesh but also from the input space near each geometry point. The input points to the first MLP projection of the GeoTransolver model are likewise enhanced with geometrical information near to them.

4 Experiment details

4.1 Datasets

The benchmarking, validation and ablation studies are carried out on 3 datasets, DrivAerML [10], Luminary SHIFT-SUV [11] and Luminary SHIFT-Wing [12]. Below we provide details about each of these datasets with information such as training, validation and testing splits.

4.1.1 DrivAerML

DrivAerML [10] is a public, high-fidelity dataset tailored for AI-based surrogates in automotive external aerodynamics. It comprises of 500 parametrically morphed variants of the DrivAer Notchback, providing broad geometric diversity for studying drag and flow behavior at scale. Each case is simulated with a scale-resolving hybrid RANS/LES approach representative of industry practice [41, 42, 43, 44], on extremely large meshes—on the order of 140 to 150 million volume elements and roughly 9 to 10 million surface points/elements [45, 10].

For every geometry, time-averaged fields are released in VTK formats: VTP files contain surface quantities (pressure and wall shear stress), and VTU files contain volume fields (velocity, pressure, and turbulence-related variables such as vorticity or turbulent viscosity). The geometry is exported as a coarse STL with around 0.3 million points. The dataset was created to address the lack of open-source, large-scale CFD data suitable for high-fidelity ML research in automotive aerodynamics.

In our study, we adopt a drag-aware split strategy. 10 % of the samples are held out for testing, and approximately 20% of this test set is designated out-of-distribution (OOD) based on drag ranges. These OOD cases correspond to some of the lowest and highest drag configurations and are not exposed during training, enabling assessment of generalization to extreme geometries and regimes. Details related to the exact training and validation split may be found in PhysicsNeMo [8] here: https://github.com/NVIDIA/physicsnemo-cfd/tree/main/workflows/bench_example/drivaer_ml_files.

4.1.2 Luminary SHIFT-SUV

SHIFT-SUV is an open-source, high-fidelity database of external aerodynamics developed by Luminary Cloud in collaboration with Honda, comprising thousands of transient simulations of parametrically morphed variants of the AeroSUV platform from FKFS (Forschungsinstitut für Kraftfahrwesen und Fahrzeugmotoren Stuttgart) [46]. AeroSUV shares design lineage and geometric characteristics with the widely used DrivAer platform in sedan aerodynamics [47]. Distributed under a CC-BY-NC license and hosted on HuggingFace [11], SHIFT-SUV is purpose-built for high-fidelity aerodynamic inference without requiring CFD or meshing expertise, supporting both surface- and volume-based surrogate training, real-time inference, and exploration of shape–performance correlations.

Geometry generation uses a deformation-cage approach implemented in ANSA (BETA-CAE Systems). Cage vertex motions are structured to emulate vehicle design parameters familiar to stylists and defined with guidance from Honda; the parameters are non-orthogonal (multiple parameters may affect shared control points). Configuration options—such as body style (Estate vs. Fastback) and underbody detail (smooth vs. detailed)—are treated as non-parametric toggles. Regions that should not vary across designs (e.g., wheels, tires, suspension) are excluded from morphing to preserve consistency. Specific parameter values are selected via Latin hypercube sampling. The initial dataset varies geometry while keeping boundary conditions fixed. The deformation space uses a morphing-cage workflow to apply stylist relevant surface translations relative to the baseline model, with ranges specified as target displacements in millimeters that guide, but do not strictly enforce, the final geometry.

The geometric parameters are varied as follows. Because the morphing cage couples nearby regions, these bounds serve as design targets for the cage control points rather than exact, per-point displacement limits on the surface. Additional details may be found in the dataset provided by Luminary et al [11].

- Front end changes include hood height (-50 to $+50$ mm), front (FR) overhang (-150 to $+150$ mm), windshield angle implemented via forward/backward shifts (-150 to $+100$ mm), and planview adjustments near the front (-75 to $+75$ mm),
- Global and chassis-related shifts cover overall vehicle height (-150 to $+150$ mm) and ride height (-30 to $+30$ mm),
- Rear end modifications include backlight angle via surface translation (-100 to $+200$ mm), rear (RR) overhang (-150 to $+100$ mm), and tapering at both the windshield and rear (-100 to $+100$ mm and -90 to $+70$ mm, respectively).

Variants are organized into four groups by scale and body style—full-scale Estate, full-scale Fastback, quarter-scale Estate, and quarter-scale Fastback—with 998 samples in each group. Quarter-scale cases validate the setup against FKFS wind-tunnel experiments for a 1/4-scale model [48], while full-scale cases follow the same methodology with refinements informed by best practices from AutoCFD [49].

Each sample has a unique volume mesh created in ANSA, yielding hex-dominant and polyhedral meshes. A grid refinement study for the validation configuration selected settings of approximately 45 million cells as a balance between accuracy and cost. All cases were run on the Luminary Cloud platform using a GPU-native, second-order (space and time) finite-volume solver [50, 49]. The turbulence model is transient delayed detached eddy simulation (DDES)—a scale-resolving DES variant—with a shear-layer-adapted length scale and a vortex tilting measure (VTM) to mitigate grey-area effects. The Spalart–Allmaras (SA) model is employed in RANS regions, and a hybrid centered/upwind convective scheme with proprietary blending limits dissipation in LES regions. An advanced shielding function prevents modeled stress depletion, avoiding premature separation. Boundary conditions include a rolling road (translating floor) and rotating wheels, with all other vehicle surfaces treated as no-slip walls. For the full-scale datasets—used in our training—the inflow is a uniform 30 m/s [50, 49].

For training we use 1996 simulations from the full-scale dataset split randomly into 80/10/10. We match the exact simulations in each split with the AB-UPT benchmarking work in Alkin et al. [51].

4.1.3 Luminary SHIFT-Wing

SHIFT-Wing is an open-source database centered on the NASA Common Research Model (CRM) [52] for high-speed transonic transport aerodynamics [12]. The CRM spans a wide range of configurations—from cruise to high-lift with deployed flaps/slats, optional nacelles/pylons, and empennage—and has been extensively studied experimentally and through community CFD efforts (e.g., AIAA Drag Prediction and High-Lift workshops). Developed in collaboration with Otto Aviation, the dataset is purpose-built for high-fidelity aerodynamic inference of non-linear flow fields without requiring CFD or meshing expertise, supporting both surface- and volume-based surrogate training, real-time inference, and exploration of shape–performance trade-offs.

The current release focuses on the high-speed cruise configuration with only fuselage and wing, emphasizing planform design. A fully parametric CRM was constructed in OnShape by importing the NASA reference, deconstructing it, and reassembling it with exposed variables. The parameterization separates wing and fuselage and proceeds as follows:

- Wing: intersect the reference wing at six spanwise stations to extract airfoil profiles; expose parameters controlling profile translation and rotation (twist); re-loft the wing and reconstruct the tip geometry.
- Fuselage: parameterize fuselage length and radius; size the wing–body fairing from the local chord of the newly lofted wing.
- Assembly: combine fuselage and wing via Boolean operations. While many micro-parameters are used internally, the dataset varies a compact set of macro planform/fuselage parameters (seven in the current release), with micro-parameters derived from these.

Datasets are generated in batches at fixed Mach numbers; geometry parameters and angle of attack are selected via Latin hypercube sampling to span distinct flow regimes intentionally. Lower-Mach batches avoid shocks, whereas higher-Mach batches exhibit complex three-dimensional shock structures, enabling targeted evaluation of AI/ML methods across qualitatively different physics. The SHIFT-Wing design space targets classical wing planform and fuselage variables around the NASA CRM reference.

The design and operating parameters are varied as follows. Additional details maybe found in the dataset provided by Luminary et al [12].

- On the fuselage, aspect ratio varies from 7.5 to 11 with a reference value of 9. The quarter-chord sweep angle spans 25 to 37.5 (reference 35). The root-chord extension factor ranges from 1.0 to 1.4 (reference 1.373). Fuselage diameter ranges from 240 to 258 inches, with 240 inches as the reference.
- On the wing, root twist varies from 3 to 9 (reference 6.717). Spanwise twist is controlled via deltas: root-to-break ranges from -7 to -3 (reference -5.953), and break to tip ranges from -7.5 to -1.5 (reference -4.513).
- Operating conditions are varied through angle of attack from 0 to 4 and Mach number from 0.5 to 0.85.

All simulations are run on the Luminary Cloud platform using a GPU-native finite-volume solver. For SHIFT-Wing, turbulence is modeled via steady RANS with the Spalart–Allmaras model [50]. All vehicle surfaces are treated as no-slip walls, and angle of attack is imposed through far-field boundary conditions. A key distinction from SHIFT-SUV is meshing: SHIFT-Wing employs Luminary Mesh Adaptation (LMA), a proprietary solution-adaptive meshing workflow that iteratively refines local mesh density and anisotropy to capture sharp features (e.g., transonic shocks) without user intervention. This automated sequence of meshes and solutions improves accuracy across diverse geometries and flow conditions while eliminating manual meshing effort.

For training, we use 1698 simulations (1138 at Mach 0.5 and 560 at Mach 0.85) from this dataset split randomly into 80/10/10. We match the exact simulations in each split with the AB-UPT benchmarking work in Alkin et al. [51].

4.2 Models

We evaluate GeoTransolver on SHIFT-SUV and SHIFT-Wing alongside strong baselines—Transolver and DoMINO—and report AB-UPT results directly from Alkin et al. [51]. To probe architectural sensitivity, we also conduct an ablation study on DrivAerML. Transolver uses the implementation released in PhysicsNeMo, with full self-attention over concatenated surface and volume tokens ($\tilde{1}0M$ parameters). DoMINO is run with the default PhysicsNeMo configuration following Ranade et al. [7], including multi-scale neighborhoods and auxiliary geometric features (surface normals, areas, signed distance), yielding $\tilde{1}9.7M$ parameters. AB-UPT is not re-implemented or re-run; we adopt the refined 384-dimensional results from Alkin et al. [51]. GeoTransolver, released in PhysicsNeMo, spans $\tilde{1}0\text{--}25M$ parameters depending on GALE depth, ball-query radii, and kernel size; Section 5.3 reports accuracy across these architectural choices on DrivAerML. For SHIFT-Wing, GeoTransolver conditions each block on global parameters (angle of attack, Mach) via geometry/global context projections, in contrast to Transolver’s plain token conditioning. All models are trained for up to 500 epochs on a single NVIDIA GB200 node using the Muon optimizer [53], under a shared preprocessing and evaluation protocol.

4.3 Metrics

The prediction accuracy is compared across the metrics mentioned below and averaged over test samples. j refer to the index of the test sample and $\tilde{\cdot}$ denotes the model prediction.

- Mean Absolute Error:

$$\frac{1}{N} \sum_{j=1}^N \|\mathbf{u}_j - \tilde{\mathbf{u}}_j\| \quad (15)$$

- Relative L_1 Norm:

$$\frac{\sum_{j=1}^N \|\mathbf{u}_j - \tilde{\mathbf{u}}_j\|}{\sum_{j=1}^N \|\mathbf{u}_j\|} \quad (16)$$

For surface predictions, we also compare lift and drag forces on the test geometries. These global metrics are commonly used during design exploration for industrial engineering applications. The total force vector on test geometries is obtained as follows:

$$\mathbf{F} = \oint_S (-(p_s - p_\infty) \hat{\mathbf{n}} + \boldsymbol{\tau}_w) dS, \quad (17)$$

where, p_s denotes the surface pressure, $\boldsymbol{\tau}_w$ is the shear stress, $\hat{\mathbf{n}}$ is the normal vector. For calculating the drag force, the normal vector is set in the tangential direction, $n = [1, 0, 0]$, while for lift force it is in the direction perpendicular to

flow, $n = [0, 0, 1]$. The lift and drag forces are computed for each of the test cases using both predicted and ground truth fields and the R^2 value is calculated over the entire test set, to determine the predictive accuracy.

For the best-performing GeoTransolver configuration, we provide qualitative assessments: contour plots of key fields (e.g., pressure coefficient, velocity magnitude) and design trends showing predicted drag/lift as functions of representative geometry or regime parameters, alongside ground-truth curves where available.

5 Results and Discussions

5.1 DrivAerML

Table 1 reports relative L_1 errors % between GeoTransolver predictions and ground truth on the DrivAerML test set (48 designs). Model hyperparameters are selected via the ablation study in Section 5.3. With the optimal configuration, field reconstructions on both surface and volume are below 5%, and the integrated aerodynamic coefficients computed from surface fields closely match ground truth (drag/lift with near-perfect agreement).

Table 1: Relative L1 errors (in %) on DrivAerML test geometries. The best GeoTransolver architecture is determined from the ablation study experiments presented in 5.3.

Model	Surface				Volume	
	p_s	τ_w	C_D	C_L	p_v	u
GeoTransolver	2.86	4.9	0.996	0.991	3.09	4.02

5.2 DrivAerML design trends

Figure 2 shows drag and lift coefficients across the 48 test designs, with designs ordered by ascending ground-truth values. The predicted curves closely track the ground-truth trends, preserving the overall ordering and capturing large-scale variations. Small oscillations appear between adjacent designs with subtle directional changes, indicating minor non-monotonic deviations in tightly clustered regions. The design points with the highest and lowest drag and lift forces correspond to out-of-distribution and are never seen during training. It may be noted that the model captures the aerodynamic forces for these designs reasonably accurately and the predictions can be further improved with augmentation of the training set.

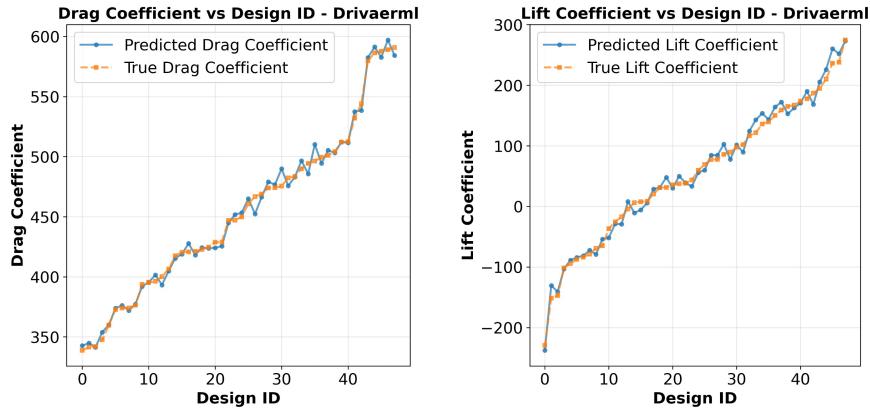


Figure 2: Design trends, Drag and Lift vs Test Designs

5.3 DrivAerML: Ablation Studies

This section provides a detailed ablation analysis of the GeoTransolver across variations in different hyperparameters such as layers, token sizes and multiscale ball-query features.

5.3.1 GALE layers

Table 2: Relative L_1 errors (in %) on DrivAerML test geometries for number of GALE layers.

Layers	Params	Surface				Volume	
		p_s	τ_w	C_D	C_L	p_v	u
6	9M	3.52	5.88	0.996	0.987	3.79	4.44
10	14M	3.25	5.51	0.995	0.987	3.31	4.21
14	20M	3.11	5.29	0.995	0.983	3.34	4.24
18	26M	2.95	5.19	0.994	0.987	3.08	4.08
20	29M	2.86	4.9	0.996	0.991	3.09	4.02

Table 2 reports how GALE depth affects accuracy on DrivAerML. As the number of GALE layers increases from 6 (9M params) to 20 (29M params), relative L_1 errors on fields generally decrease: surface pressure (p_s) drops from 3.52% to 2.86% (19% relative improvement), wall shear stress (τ_w) from 5.88% to 4.90% (17%), volume pressure (p_v) from 3.79% to 3.09% (19%), and velocity (v) from 4.44% to 4.02% (9%). Drag and lift coefficients, reported as R_2 rather than L_1 , are high throughout ($C_D \approx 0.994$ to 0.996, $C_L \approx 0.983$ to 0.991) with a modest gain in lift at larger depths. There is mild non-monotonicity at 14 layers (slightly higher volume errors than at 10), but performance recovers and improves at 18 and peaks at 20 layers, which achieves the best results across all metrics in the table. Overall, deeper GALE stacks yield consistent accuracy gains at the cost of increased parameters, with diminishing but still positive returns near 18–20 layers.

5.3.2 Ball Query multi-scale features

Table 3: Relative L1 errors (in %) on DrivAerML test geometries for ball query features

Multi-scale radii	Params	Surface				Volume	
		p_s	τ_w	C_D	C_L	p_v	u
0.05	12M	3.14	5.38	0.993	0.989	3.6	4.34
2.5	12M	3.09	5.38	0.995	0.986	3.24	4.20
0.05, 0.25, 1.0, 2.5	21M	3.03	5.23	0.993	0.989	3.06	4.06
0.01, 0.05, 0.25, 1.0, 2.5, 5.0	29M	2.86	4.9	0.996	0.991	3.09	4.02

Table 3 examines how the choice of ball-query radii affects GeoTransolver’s accuracy on the DrivAerML dataset. Using a single small neighborhood ($r=0.05$) yields baseline errors, p_s of 3.14%, τ_w of 5.38%, p_v of 3.60%, v of 4.34% with strong drag/lift R_2 , C_D of 0.993 and C_L of 0.989. A single large neighborhood ($r=2.5$) modestly improves field errors, p_s of 3.09%, p_v of 3.24%, u of 4.20% at the same parameter count (12M). Introducing four scales ($r=0.05, 0.25, 1.0, 2.5$; 21M parameters) delivers broader gains across surface and volume, p_s of 3.03%, τ_w of 5.23%, p_v of 3.06%, u of 4.06%; C_D of 0.993, C_L of 0.989. The richest six-scale setting (0.01, 0.05, 0.25, 1.0, 2.5, 5.0; 29M params) achieves the best overall performance—lowest surface errors, p_s of 2.86%, τ_w of 4.90%, strongest coefficients C_D of 0.996 and C_L of 0.991, and best velocity error, u of 4.02% — with volume pressure essentially on par with the four-scale case (3.09% vs 3.06%). Overall, expanding from single- to multi-scale neighborhoods consistently improves accuracy, with finer radii aiding near-wall detail and larger radii capturing global couplings; returns diminish slightly at the highest capacity but remain positive across most metrics.

5.3.3 Ball Query kernel size

Table 4: Relative L1 errors (in %) on DrivAerML test geometries for ball query features

kernel size	Params	Surface				Volume	
		p_s	τ_w	C_D	C_L	p_v	u
8	13M	3.12	5.32	0.994	0.984	3.37	4.06
16	18M	3.07	5.27	0.993	0.989	3.12	4.11
32	29M	2.86	4.9	0.996	0.991	3.01	4.02

Table 4 studies the effect of ball-query kernel size (maximum neighbors per radius) on DrivAerML accuracy. Increasing the kernel from 8 (13M params) to 16 (18M) yields modest gains in p_s (3.12% to 3.07%) and τ_w of (5.32% to 5.27%), and a clearer improvement in p_v (3.37% to 3.12%); velocity error is roughly flat (4.06% to 4.11%) and drag/lift R_2 remains high (0.994/0.984 to 0.993/0.989). At 32 neighbors (29M), the model achieves the best results across all metrics: p_s of 2.86%, τ_w of 4.90%, p_v of 3.01%, u of 4.02%, with drag/lift R_2 at 0.996/0.991. Overall, larger kernels improve accuracy by enriching local geometric aggregation, with small, non-monotonic fluctuations at intermediate size, at the cost of increased parameters and compute.

5.3.4 Geometry and Query tokens

Table 5: Relative L1 errors (in %) on DrivAerML test geometries for number of geometry and query points.

Query/Geo tokens	Surface				Volume	
	p_s	τ_w	C_D	C_L	p_v	u
20k/50k	3.08	5.36	0.993	0.989	11.1	9.3
20k/150k	3.02	5.29	0.994	0.989	7.03	6.74
20k/300k	3.04	5.27	0.993	0.987	3.09	4.01
40k/50k	3.03	5.22	0.993	0.985	3.01	3.97
40k/150k	2.96	5.20	0.995	0.988	3.13	3.97
40k/300k	2.98	5.25	0.994	0.987	4.41	4.76
60k/50k	2.97	5.17	0.994	0.987	5.61	5.75
60k/150k	2.99	5.15	0.994	0.987	2.96	4.02
60k/300k	2.86	4.9	0.996	0.991	3.01	4.02

Table 5 evaluates how the number of query tokens and geometry tokens used by the context builder impacts accuracy on the DrivAerML dataset. Increasing geometry token density at fixed queries sharply reduces volume errors early on (e.g., at 20k queries, p_v drops from 11.1% at 50k geo to 3.09% at 300k), indicating that adequate geometric coverage is critical for stable volume reconstruction. Scaling query tokens also improves surface metrics: moving from 20k to 60k queries systematically lowers surface pressure and wall-shear errors, with the best surface and coefficient performance at 60k/300k (p_s of 2.86%, τ_w of 4.90%, C_D of 0.996, C_L of 0.991). Volume metrics exhibit a sweet spot: 60k/150k attains the lowest p_v (2.96%) with competitive velocity error (u 4.02%), while 40k/50k already yields strong u (3.97%) with low p_v (3.01%). At very high geometry counts paired with moderate queries (e.g., 40k/300k), volume errors can degrade, suggesting an imbalance between query capacity and geometry tokens. Overall, balanced increases in both query and geometry tokens improve accuracy, with diminishing and occasionally non-monotonic returns; 60k/300k is best overall for surface and integral metrics, whereas 60k/150k slightly edges volume pressure.

5.4 SHIFT-SUV Results

5.4.1 Relative L_1 errors

Table 6 below compares the relative L_1 error across different model architectures for both Estate and Fastback test geometries. We observe that GeoTransolver yields better accuracy than other model architectures in most of the metrics except volumetric pressure prediction for the Estate where it is slightly lower than AB-UPT.

Table 6: Relative L1 errors (in %) on Estate and Fastback test geometries

Model	Estate				Fastback			
	Surface		Volume		Surface		Volume	
	p_s	τ_w	p_v	u	p_s	τ_w	p_v	u
GeoTransolver	0.0057	3.81	0.0026	1.36	0.0056	3.70	0.0023	1.30
AB-UPT	0.0064	4.95	0.0025	2.25	0.0064	5.03	0.0024	2.21
DoMINO	0.0100	12.24	0.0062	8.14	0.0100	11.74	0.0067	7.73
Transolver	0.0079	4.98	0.004	1.87	0.0078	4.97	0.0039	1.81

Likewise, we present a comparison of R^2 values for C_D and C_L predictions across all model architectures in table 7. All models yield similar R_2 score with GeoTransolver performing slightly better than the others.

Table 7: R^2 comparison among different models for drag (C_D) and lift (C_L) coefficient on SHIFT-SUV test geometries.

Model	Estate		Fastback	
	C_D	C_L	C_D	C_L
GeoTransolver	0.98	0.88	0.99	0.99
AB-UPT	0.96	0.82	0.98	0.99
DoMINO	0.67	0.56	0.84	0.70
Transolver	0.95	0.85	0.96	0.98

5.4.2 Contours and design trends for GeoTransolver

Figures 3 and 4 plot drag and lift coefficients across the test design IDs for the Estate and Fastback. The predicted trends closely follow ground truth, capturing overall ordering and directional changes. Larger oscillations are visible in Estate lift, likely due to sparser coverage in that lift regime.

Figures 5 and 6 compare surface pressure and wall-shear stress contours between GeoTransolver and ground truth. Errors are uniformly distributed and bounded, indicating high accuracy; the largest discrepancies occur near wheels and mirrors, where flow is particularly complex.

Similarly, Figures 7 and 8 show XZ-plane volume comparisons. The highest errors appear near separation regions downstream and in the wake where gradients are sharp, but overall predictions remain accurate.

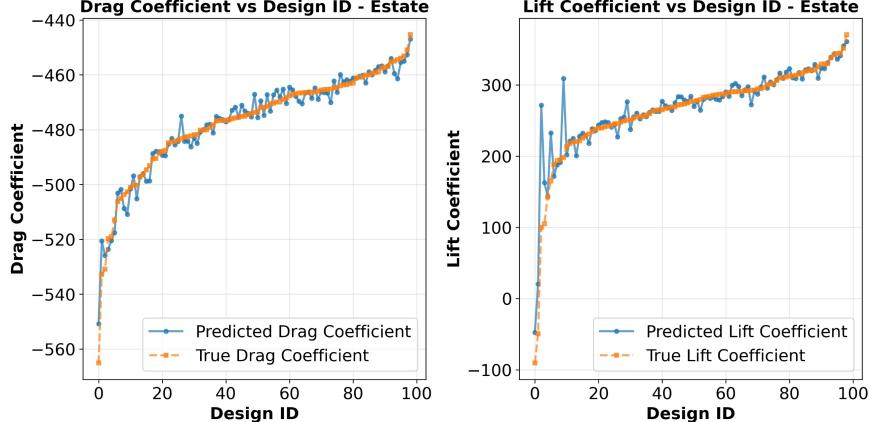


Figure 3: Design trends for Estate, Drag and Lift vs Test Designs

5.5 SHIFT-WING Results

5.5.1 Relative L_1 errors

Table 8 below compares the relative L_1 error across different model architectures for test geometries across both Mach ranges of 0.5 and 0.85. We observe that GeoTransolver yields a better accuracy than other model architectures in most of the metrics except surface pressure predictions for Mach 0.85 and wall-shear stress prediction for Mach 0.5 where AB-UPT and DoMINO perform better, respectively. For this experiment, DoMINO is trained separately on Mach 0.5 and 0.85 while rest of the models are trained on the combined dataset.

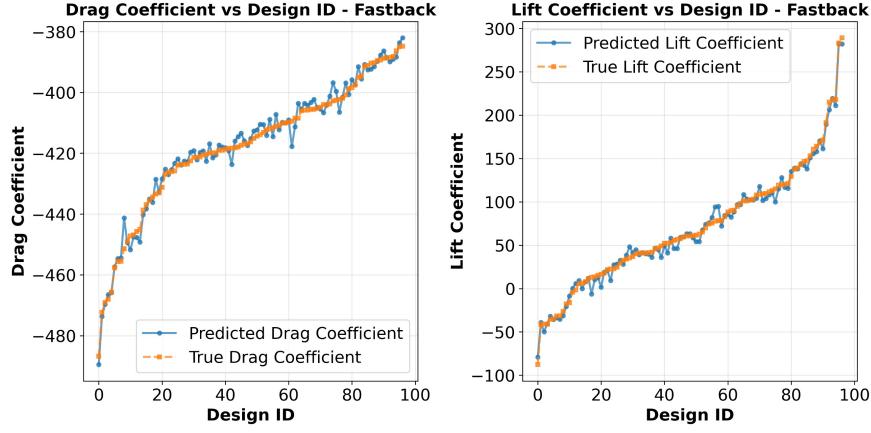


Figure 4: Design trends for Fastback, Drag and Lift vs Test Designs

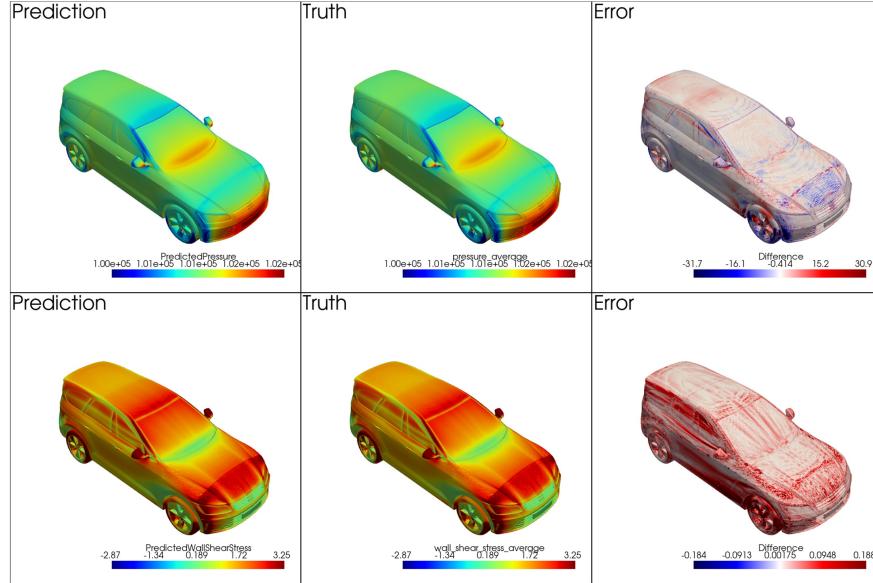


Figure 5: Surface contour comparisons for Estate

Table 8: Relative L1 errors (in %) on $Ma = 0.5$ and $Ma = 0.85$ test geometries

Model	$Ma = 0.5$				$Ma = 0.85$			
	Surface		Volume		Surface		Volume	
	p_s	τ_w	p_v	u	p_s	τ_w	p_v	u
GeoTransolver	0.021	12.2	0.022	1.92	0.081	13.01	0.099	2.00
AB-UPT	0.022	12.5	0.027	9.56	0.079	13.3	0.125	9.51
DoMINO	0.468	10.2	2.25	21.34	1.88	13.35	3.17	29.2
Transolver	0.094	12.4	1.19	3.60	0.098	13.2	1.34	3.72

Likewise, we present a comparison of R^2 values for C_D and C_L predictions across all model architectures in table 9. All models perform exceptionally well and provide similar accuracy for this metric.

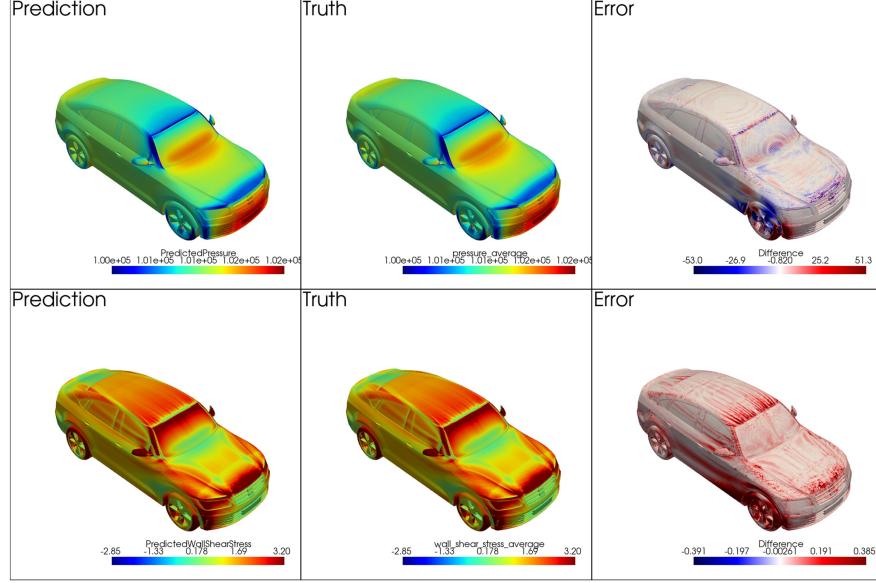


Figure 6: Surface contour comparisons for Fastback

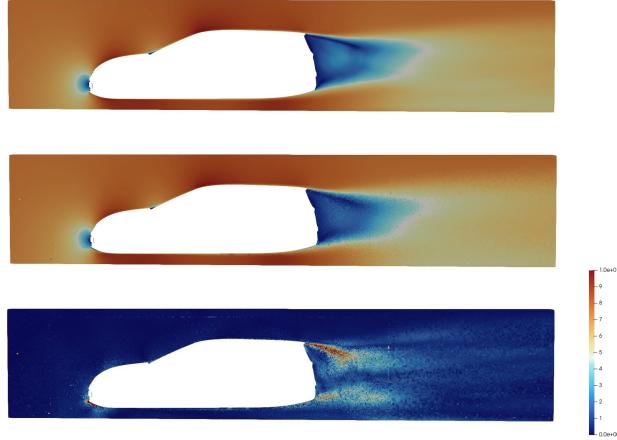


Figure 7: Volume contour comparisons for Estate on XZ plane

Table 9: R^2 comparison among different models for drag (C_D) and lift (C_L) coefficient on SHIFT-WING test geometries.

Model	$Ma = 0.5$		$Ma = 0.85$	
	C_D	C_L	C_D	C_L
GeoTransolver-MAE	1.0	1.0	1.0	1.0
AB-UPT	1.0	1.0	1.0	1.0
DoMINO	1.0	1.0	1.0	1.0
Transolver	1.0	1.0	1.0	1.0

5.5.2 Contours and design trends for GeoTransolver

Figures 9 and 10 plot drag and lift across test design IDs at Mach 0.5 and 0.85. Predicted trends closely track ground truth in both regimes, preserving ordering and directional changes.

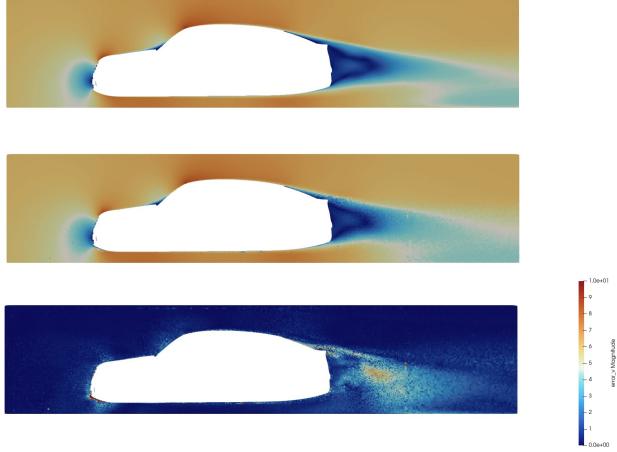


Figure 8: Volume contour comparisons for Fastback on XZ plane

Figures 11 and 12 compare surface pressure and wall-shear stress contours between GeoTransolver and ground truth. Errors are uniformly distributed and bounded, indicating high accuracy; the largest discrepancies appear on the wing, especially at higher Mach where sharper transitions occur.

Figures 13 and 14 show XZ-plane volume comparisons. Errors are most pronounced over the wing and fuselage and in the downstream wake where separation and steep gradients occur, particularly at Mach 0.85. Overall, predictions remain accurate across both regimes.

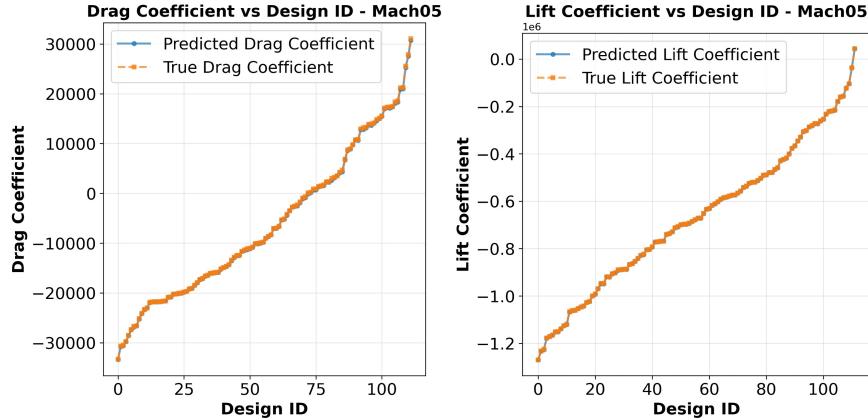


Figure 9: Design trends for Mach 0.5, Drag and Lift vs Test Designs

6 Conclusion

In this work, we introduced GeoTransolver, a novel transformer architecture for CAE that builds on the Transolver architecture by integrating Geometry-Aware Latent Embeddings (GALE). By replacing standard attention mechanisms with GALE, we successfully coupled physics-aware self-attention on learned state slices with cross-attention to a global, geometry-aware context. This approach, underpinned by multi-scale ball queries inspired by Domino, ensures that latent computations are persistently anchored to the underlying domain structure and operating regimes at every block. We extensively benchmarked GeoTransolver on the DrivAerML, SHIFT-SUV, and SHIFT-Wing datasets. Our results demonstrate that GeoTransolver consistently delivers competitive accuracy compared to state-of-the-art baselines, including Domino, Transolver, and AB-UPT. Notably, the model exhibits improved robustness to geometric and regime shifts while maintaining favorable data efficiency. By unifying geometry-aware context with scalable transformer attention, GeoTransolver effectively bridges the gap between the flexibility of unstructured geometric deep learning and the long-range reasoning capabilities of transformers.

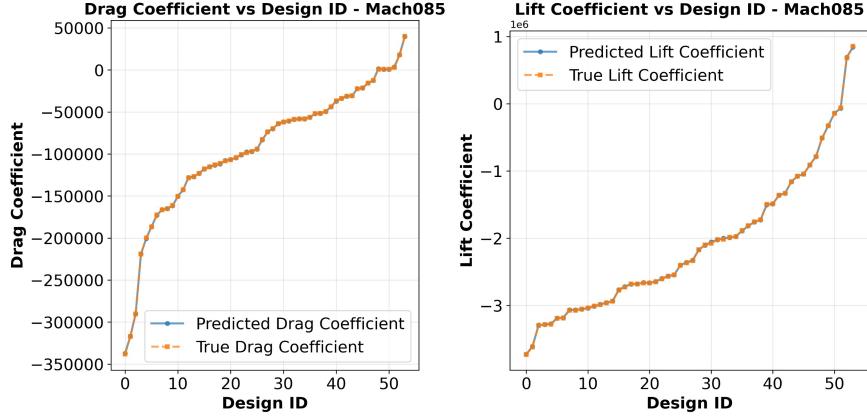


Figure 10: Design trends for Mach 0.85, Drag and Lift vs Test Designs

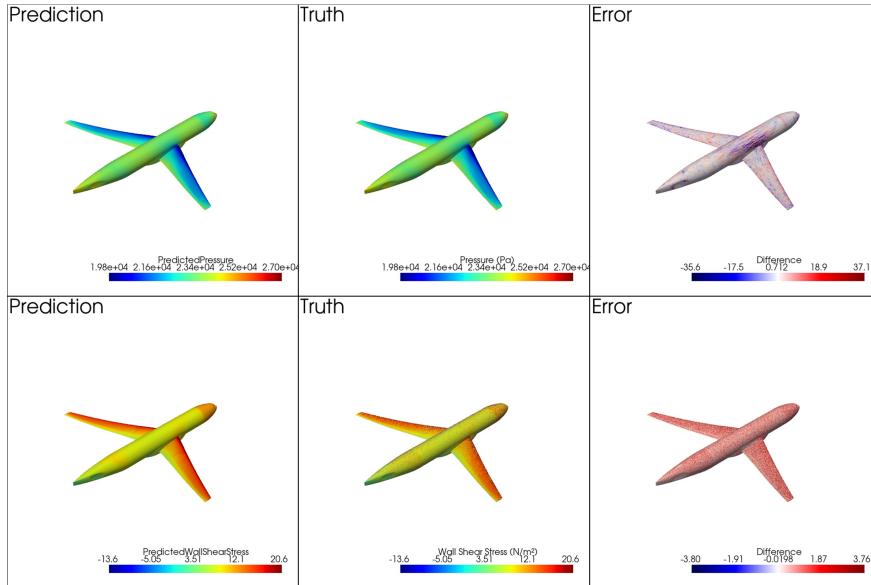


Figure 11: Surface contour comparisons for Mach 0.5

6.1 Broader impact

The development of GeoTransolver architecture has significant implications for the field of CAE. First, we have shown that by constructing high quality geometric and global contexts, transformer-like architectures such as Transolver may be significantly enhanced with cross attention to ensure AI surrogate solvers retain geometrical information throughout the model. Our experiments showed, in a direct comparison with Transolver, that this cross attention enhances the model’s predictive power.

Further, the high-fidelity creation of the context vector for GeoTransolver and related ablation experiments show just how impactful properly embedding geometrical information is. We demonstrated that not just spatial resolution but multi-scale features enhance the predictive power of GeoTransolver.

Geometrical and boundary-condition context is critical to the generalization of CFD AI surrogate solvers. The demonstrations here represents a significant step towards foundational models for CFD AI surrogate solvers: not only does this model equal or exceed state of the art performance on benchmark datasets, we believe the concepts demonstrated herein will be important for the construction of highly-generalizable surrogate solvers in the near future.

GeoTransolver is also released in an open source software package, PhysicsNeMo, making it accessible and usable in part or in whole to all CFD researchers and application engineers.

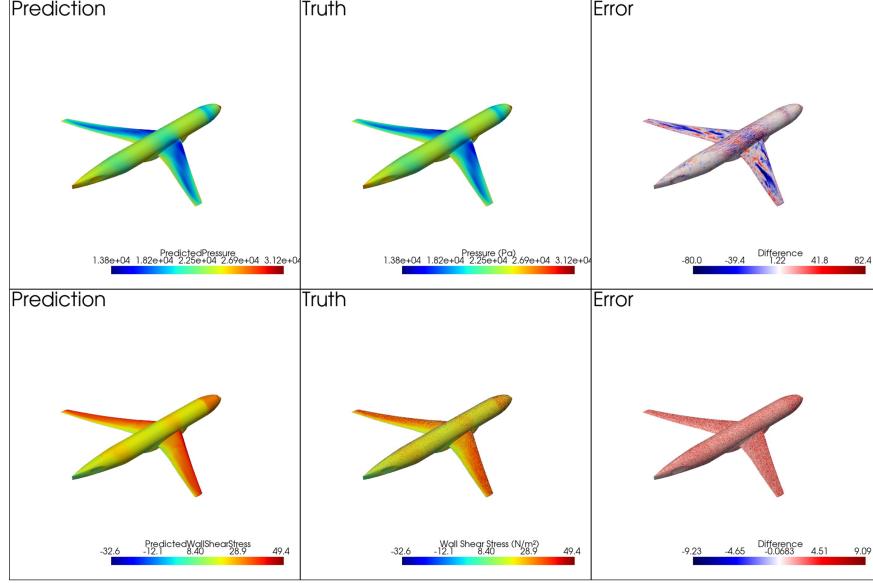


Figure 12: Surface contour comparisons for Mach 0.85

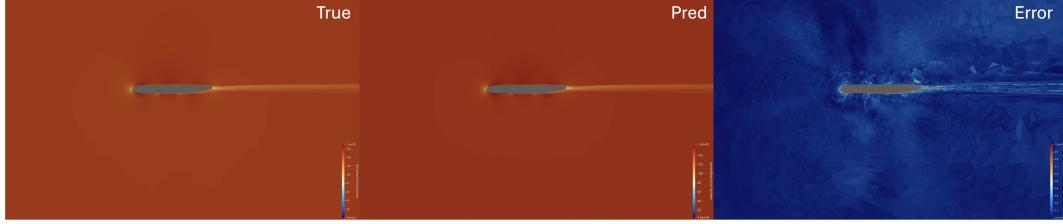


Figure 13: Volume contour comparisons on XZ plane for Mach 0.5

6.2 Future work

While GeoTransolver demonstrates strong performance on fluid dynamic workloads, several avenues for future research remain:

- **Physics Constrained Training:** Although the current architecture supports constraint-aware training, future iterations could explicitly integrate physics based losses to enforce PDE constraints (e.g., incompressibility) and boundary conditions more strictly during training. Further, we may seek to adapt the data pre-processing and model architecture to better preserve equivariances and invariances in the underlying symmetries of the systems.
- **Validate on Multi-Physics problems:** Leveraging the training recipes in PhysicsNeMo, we aim to extend GeoTransolver to coupled problems, such as conjugate heat transfer or aeroacoustics, where the interaction between diverse physical fields and complex geometry is critical.
- **Extend to Design Optimization workflows:** We plan to integrate GeoTransolver directly into gradient-based design optimization workflows. The differentiable nature of the architecture makes it an ideal candidate for end-to-end inverse design, moving beyond prediction to automated shape optimization.



Figure 14: Volume contour comparisons on XZ plane for Mach 0.85

References

- [1] Xu-Wen Zhao, Xiao-Meng Tong, Fang-Wei Ning, Mao-Lin Cai, Fei Han, and Hong-Guang Li. Review of empowering computer-aided engineering with artificial intelligence. *Advances in Manufacturing*, pages 1–41, 2025.
- [2] Sofos Filippou. Advancing fluid mechanics with artificial intelligence and machine learning. *Fluids*, 10(11):297, 2025.
- [3] Steven L Brunton, Bernd R Noack, and Petros Koumoutsakos. Machine learning for fluid mechanics. *Annual review of fluid mechanics*, 52(1):477–508, 2020.
- [4] Pushan Sharma, Wai Tong Chung, Bassem Akoush, and Matthias Ihme. A review of physics-informed machine learning in fluid mechanics. *Energies*, 16(5):2343, 2023.
- [5] Neil Ashton, Johannes Brandstetter, and Siddhartha Mishra. Fluid intelligence: A forward look on ai foundation models in computational fluid dynamics. *arXiv preprint arXiv:2511.20455*, 2025.
- [6] Haixu Wu, Huakun Luo, Haowen Wang, Jianmin Wang, and Mingsheng Long. Transolver: A fast transformer solver for pdes on general geometries. *arXiv preprint arXiv:2402.02366*, 2024.
- [7] Rishikesh Ranade, Mohammad Amin Nabian, Kaustubh Tangsali, Alexey Kamenev, Oliver Hennigh, Ram Cherukuri, and Sanjay Choudhry. Domino: A decomposable multi-scale iterative neural operator for modeling large scale engineering simulations. *arXiv preprint arXiv:2501.13350*, 2025.
- [8] NVIDIA. NVIDIA PhysicsNeMo, 2025.
- [9] Benedikt Alkin, Maurits Bleeker, Richard Kurle, Tobias Kronlachner, Reinhard Sonnleitner, Matthias Dorfer, and Johannes Brandstetter. Ab-up: Scaling neural cfd surrogates for high-fidelity automotive aerodynamics simulations via anchored-branched universal physics transformers. *arXiv preprint arXiv:2502.09692*, 2025.
- [10] Neil Ashton, Charles Mockett, Marian Fuchs, Louis Fliessbach, Hendrik Hetmann, Thilo Knacke, Norbert Schonwald, Vangelis Skaperdas, Grigoris Fotiadis, Astrid Walle, et al. Drivaerml: High-fidelity computational fluid dynamics dataset for road-car external aerodynamics. *arXiv preprint arXiv:2408.11969*, 2024.
- [11] Luminary Cloud. Shift-suv: High-fidelity computational fluid dynamics dataset for suv external aerodynamics. *URL https://huggingface.co/datasets/luminary-shift/SUV/*, 2025.
- [12] Luminary Cloud. Shift-wing: High-fidelity computational fluid dynamics dataset for transonic wing external aerodynamics. *URL https://huggingface.co/datasets/luminary-shift/WING/*, 2025.
- [13] Kaustubh Tangsali, Rishikesh Ranade, Mohammad Amin Nabian, Alexey Kamenev, Peter Sharpe, Neil Ashton, Ram Cherukuri, and Sanjay Choudhry. A benchmarking framework for ai models in automotive aerodynamics. *arXiv preprint arXiv:2507.10747*, 2025.
- [14] Nikola Kovachki, Zongyi Li, Burigede Liu, Kamyar Azizzadenesheli, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Neural operator: Learning maps between function spaces with applications to pdes. *Journal of Machine Learning Research*, 24(89):1–97, 2023.
- [15] Lu Lu, Pengzhan Jin, Guofei Pang, Zhongqiang Zhang, and George Em Karniadakis. Learning nonlinear operators via deeponet based on the universal approximation theorem of operators. *Nature machine intelligence*, 3(3):218–229, 2021.
- [16] Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. *arXiv preprint arXiv:2010.08895*, 2020.

- [17] Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Andrew Stuart, Kaushik Bhattacharya, and Anima Anandkumar. Multipole graph neural operator for parametric partial differential equations. *Advances in Neural Information Processing Systems*, 33:6755–6766, 2020.
- [18] Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Neural operator: Graph kernel network for partial differential equations. *arXiv preprint arXiv:2003.03485*, 2020.
- [19] Zongyi Li, Daniel Zhengyu Huang, Burigede Liu, and Anima Anandkumar. Fourier neural operator with learned deformations for pdes on general geometries. *Journal of Machine Learning Research*, 24(388):1–26, 2023.
- [20] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.
- [21] Zhongkai Hao, Zhengyi Wang, Hang Su, Chengyang Ying, Yinpeng Dong, Songming Liu, Ze Cheng, Jian Song, and Jun Zhu. Gnot: A general neural operator transformer for operator learning. In *International Conference on Machine Learning*, pages 12556–12569. PMLR, 2023.
- [22] Andrey Bryutkin, Jiahao Huang, Zhongying Deng, Guang Yang, Carola-Bibiane Schönlieb, and Angelica Aviles-Rivero. Hamlet: Graph transformer neural operator for partial differential equations. *arXiv preprint arXiv:2402.03541*, 2024.
- [23] Benjamin Shih, Ahmad Peyvan, Zhongqiang Zhang, and George Em Karniadakis. Transformers as neural operators for solutions of differential equations with finite regularity. *Computer Methods in Applied Mechanics and Engineering*, 434:117560, 2025.
- [24] Qibang Liu, Weiheng Zhong, Hadi Meidani, Diab Abueidda, Seid Koric, and Philippe Geubelle. Geometry-informed neural operator transformer. *arXiv preprint arXiv:2504.19452*, 2025.
- [25] Oded Ovadia, Adar Kahana, Panos Stinis, Eli Turkel, Dan Givoli, and George Em Karniadakis. Vito: Vision transformer-operator. *Computer Methods in Applied Mechanics and Engineering*, 428:117109, 2024.
- [26] Tobias Pfaff, Meire Fortunato, Alvaro Sanchez-Gonzalez, and Peter Battaglia. Learning mesh-based simulation with graph networks. In *International conference on learning representations*, 2020.
- [27] Ugo Pelissier, Augustin Parret-Fréaud, Felipe Bordeu, and Youssef Mesri. Graph neural networks for mesh generation and adaptation in structural and fluid mechanics. *Mathematics*, 12(18):2933, 2024.
- [28] Pourya Pilva and Ahmad Zareei. Learning time-dependent pde solver using message passing graph neural networks. *arXiv preprint arXiv:2204.07651*, 2022.
- [29] Rini Jasmine Gladstone, Helia Rahmani, Vishvas Suryakumar, Hadi Meidani, Marta D’Elia, and Ahmad Zareei. Mesh-based gnn surrogates for time-independent pdes. *Scientific reports*, 14(1):3394, 2024.
- [30] Mohammad Amin Nabian, Chang Liu, Rishikesh Ranade, and Sanjay Choudhry. X-meshgraphnet: Scalable multi-scale graph neural networks for physics simulation. *arXiv preprint arXiv:2411.17164*, 2024.
- [31] Meire Fortunato, Tobias Pfaff, Peter Wirnsberger, Alexander Pritzel, and Peter Battaglia. Multiscale meshgraphnets. *arXiv preprint arXiv:2210.00612*, 2022.
- [32] Chris Choy, Alexey Kamenev, Jean Kossaifi, Max Rietmann, Jan Kautz, and Kamyar Azizzadenesheli. Factorized implicit global convolution for automotive computational fluid dynamics prediction. *arXiv preprint arXiv:2502.04317*, 2025.
- [33] Qian Chen, Mohamed Elrefaie, Angela Dai, and Faez Ahmed. Tripnet: Learning large-scale high-fidelity 3d car aerodynamics with triplane networks. *arXiv preprint arXiv:2503.17400*, 2025.
- [34] Moshe Eliasof, Eldad Haber, and Eran Treister. Pde-gcn: Novel architectures for graph neural networks motivated by partial differential equations. *Advances in neural information processing systems*, 34:3836–3849, 2021.
- [35] Rishikesh Ranade, Chris Hill, Lalit Ghule, and Jay Pathak. A composable machine-learning approach for steady-state simulations on high-resolution grids. *Advances in Neural Information Processing Systems*, 35:17386–17401, 2022.
- [36] Wenzhuo Liu, Mouadh Yagoubi, and Marc Schoenauer. Multi-resolution graph neural networks for pde approximation. In *International Conference on Artificial Neural Networks*, pages 151–163. Springer, 2021.
- [37] Yuying Liu, Colin Ponce, Steven L Brunton, and J Nathan Kutz. Multiresolution convolutional autoencoders. *Journal of Computational Physics*, 474:111801, 2023.
- [38] John Guibas, Morteza Mardani, Zongyi Li, Andrew Tao, Anima Anandkumar, and Bryan Catanzaro. Adaptive fourier neural operators: Efficient token mixers for transformers. *arXiv preprint arXiv:2111.13587*, 2021.

- [39] Maziar Raissi, Paris Perdikaris, and George E Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019.
- [40] Zongyi Li, Hongkai Zheng, Nikola Kovachki, David Jin, Haoxuan Chen, Burigede Liu, Kamyar Azizzadenesheli, and Anima Anandkumar. Physics-informed neural operator for learning partial differential equations. *ACM/IMS Journal of Data Science*, 1(3):1–27, 2024.
- [41] Philippe R Spalart, Shur Deck, Michael L Shur, Kyle D Squires, M Kh Strelets, and Andrei Travin. A new version of detached-eddy simulation, resistant to ambiguous grid densities. *Theoretical and computational fluid dynamics*, 20(3):181–195, 2006.
- [42] Bruno Chaouat. The state of the art of hybrid rans/les modeling for the simulation of turbulent flows. *Flow, turbulence and combustion*, 99(2):279–327, 2017.
- [43] Stefan Heinz. A review of hybrid rans-les methods for turbulent flows: Concepts and applications. *Progress in Aerospace Sciences*, 114:100597, 2020.
- [44] Neil Ashton, Paul Batten, Andrew W Cary, Kevin R Holst, and Vangelis Skaperdas. Hlpw-4/gmgw-3: Hybrid rans/les technology focus group workshop summary. In *AIAA Aviation 2022 Forum*, page 3293, 2022.
- [45] Burkhard Hupertz, Neil Lewington, Charles Mockett, Neil Ashton, and Lian Duan. Towards a standardized assessment of automotive aerodynamic cfd prediction capability-autocfd 2: Ford drivaaer test case summary. Technical report, SAE Technical Paper, 2022.
- [46] Chenyi Zhang, Max Tanneberger, Timo Kuthada, Felix Wittmeier, Jochen Wiedemann, and Juliane Nies. Introduction of the aerosuv-a new generic suv model for aerodynamic research. Technical report, SAE Technical Paper, 2019.
- [47] Angelina I Heft, Thomas Indinger, and Nikolaus A Adams. Experimental and numerical investigation of the drivaaer model. In *Fluids engineering division summer meeting*, volume 44755, pages 41–51. American Society of Mechanical Engineers, 2012.
- [48] Chenyi Zhang. *Aerodynamic study on the vehicle shape parameters with respect to ground simulation*. Springer, 2021.
- [49] T Economos D Engwirda J Ho M Mara and R Mansharamani. *Luminary contribution to the autocfd4 workshop*. URL <https://autocfd.org/autocfd4/>, 2024.
- [50] Joshua Krakos, Pedro Gomes, Jonathan Ho, Michael Mara, Gonzalo Sáez, Adrien Loseille, Thomas Economos, and Juan J Alonso. Gpu-based and adaptive solution technology for the 5th aiaa high lift prediction workshop. In *AIAA SCITECH 2025 Forum*, page 0496, 2025.
- [51] Benedikt Alkin, Richard Kurle, Louis Serrano, Dennis Just, and Johannes Brandstetter. Ab-upf for automotive and aerospace applications. *arXiv preprint arXiv:2510.15808*, 2025.
- [52] Nasa common research model project. high speed common research model (crm). <https://commonresearchmodel.larc.nasa.gov/home-2/high-speed-crm/>.
- [53] Chongjie Si, Debing Zhang, and Wei Shen. Adamuon: Adaptive muon optimizer. *arXiv preprint arXiv:2507.11005*, 2025.