
DrivAerStar: An Industrial-Grade CFD Dataset for Vehicle Aerodynamic Optimization

Jiyan Qiu¹ **Lyulin Kuang**¹ **Guan Wang**²
jiyanq0430@gmail.com lkuang@nvidia.com wangguan.nantes@gmail.com

Yichen Xu^{3,1,4,5} **Leiyao Cui**^{3,4,5} **Shaotong Fu**¹
xuyc@stu.pku.edu.cn cuileiyaony@gmail.com shaotongf@nvidia.com

Yixin Zhu^{3,4,5,6} ✉ **Ruihua Zhang**¹, ✉
yixin.zhu@pku.edu.cn ritaz@nvidia.com

✉ Corresponding authors ¹ NVIDIA ² Baidu, INC. ³ Peking University
⁴ State Key Lab of General AI, Peking University
⁵ Beijing Key Laboratory of Behavior and Mental Health, Peking University
⁶ Embodied Intelligence Lab, PKU-Wuhan Institute for Artificial Intelligence

<https://drivaerstar.github.io/>

Abstract

Vehicle aerodynamics optimization has become critical for automotive electrification, where drag reduction directly determines electric vehicle range and energy efficiency. Traditional approaches face an intractable trade-off: computationally expensive Computational Fluid Dynamics (CFD) simulations requiring weeks per design iteration, or simplified models that sacrifice production-grade accuracy. While machine learning offers transformative potential, existing datasets exhibit fundamental limitations—inadequate mesh resolution, missing vehicle components, and validation errors exceeding 5%—preventing deployment in industrial workflows. We present *DrivAerStar*, comprising 12,000 industrial-grade automotive CFD simulations generated using STAR-CCM+[®] software. The dataset systematically explores three vehicle configurations through 20 Computer Aided Design (CAD) parameters via Free Form Deformation (FFD) algorithms, including complete engine compartments and cooling systems with realistic internal airflow. *DrivAerStar* achieves wind tunnel validation accuracy below 1.04%—a five-fold improvement over existing datasets—through refined mesh strategies with strict wall y^+ control. Benchmarks demonstrate that models trained on this data achieve production-ready accuracy while reducing computational costs from weeks to minutes. This represents the first dataset bridging academic machine learning research and industrial CFD practice, establishing a new standard for data-driven aerodynamic optimization in automotive development. Beyond automotive applications, *DrivAerStar* demonstrates a paradigm for integrating high-fidelity physics simulations with Artificial Intelligence (AI) across engineering disciplines where computational constraints currently limit innovation.

1 Introduction

Aerodynamic optimization has become increasingly crucial as the automotive industry undergoes rapid transformation toward environmentalization, electrification, and intelligentization (Bibra et al., 2022). Precise aerodynamic design directly impacts electric vehicle range (Huluka and Kim, 2019), fuel efficiency (Abinesh and Arunkumar, 2014), fluid-excited noise (Wang et al., 2013), and high-speed stability (Brandt et al., 2022). Beyond industrial applications, high-fidelity external flow field data enables deeper integration between CFD and data-driven methods for studying complex flow

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Track on Datasets and Benchmarks.

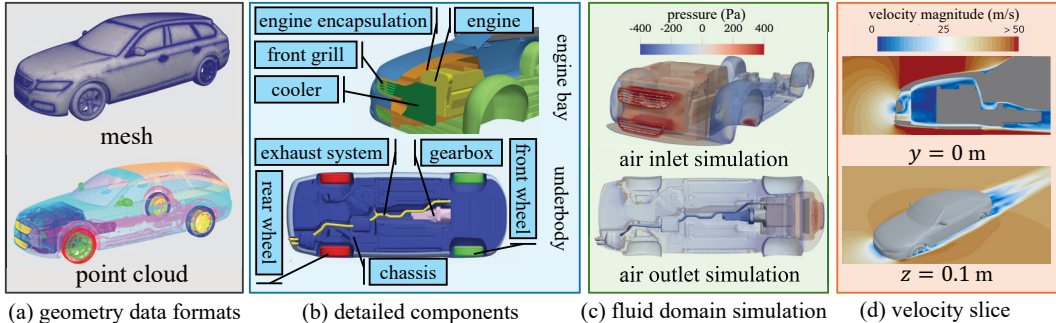


Figure 1: DrivAerStar **dataset features**. (a) Multiple geometry data formats: mesh (top) serves as primary representation while point cloud (bottom) enables alternative computational approaches and geometry analysis. (b) Complete vehicle modeling includes detailed engine bay compartment assembly (top) and underbody components (bottom), distinguishing DrivAerStar from previous automotive datasets. (c) Internal airflow simulation capabilities featuring air inlet (top) and outlet (bottom) simulations, enabled by precise internal component modeling to support comprehensive automotive system research. (d) Velocity field cross-sections showing internal engine compartment flow at vehicle centerline ($y = 0$ m) and external flow field at tire axle plane ($z = 0.1$ m).

dynamics. However, traditional numerical simulations require prohibitive computational resources, while existing datasets lack engineering fidelity for Internal Combustion Vehicle (ICV) (Payri et al., 2015), Hybrid Electric Vehicle (HEV) (Hannan et al., 2014), and Battery Electric Vehicle (BEV) (Upadhyaya and Mahanta, 2023; Bjerkan et al., 2016) applications, creating urgent demand for datasets combining precision with comprehensive parameter coverage.

Recent automotive datasets like DrivAerNet++ (Elrefaie et al., 2024) and DrivAerML (Ashton et al., 2024c) provide valuable foundations but exhibit critical limitations preventing industrial adoption. Their mesh strategies inadequately resolve gradient-dependent fields, causing boundary layer distortion and numerical diffusion errors. These resources focus on limited vehicle model deformations, failing to represent aerodynamic characteristics across diverse vehicle types and operating conditions. Furthermore, they deviate from industrial standards in turbulence modeling and solver configuration, producing results that diverge significantly from wind tunnel measurements. These shortcomings prevent the development of industrial-grade surrogate models capable of optimizing design features within minutes rather than weeks.

We present DrivAerStar, addressing these limitations through industrial-grade STAR-CCM+® simulations. Our approach overcomes OpenFOAM®’s inherent wall shear stress calculation deficiencies by implementing refined mesh strategies with targeted aerodynamic sensitivity refinements, sub-regional surface adaptation, and optimized boundary layer resolution. Strict control of dimensionless wall distance y^+ values (Schlichting and Kestin, 1961) ensures accurate boundary layer simulation while achieving superior computational precision with comparable mesh counts. The dataset spans multiple vehicle platforms with systematic FFD-based geometric parameter variations, representing diverse aerodynamic characteristics across vehicle types and operating conditions. Critically, we include complete front compartment assemblies with engine components and cooling systems, featuring realistic airflow channels that capture real-world vehicle aerodynamics.

Our 1,080,000 CPU core-hour computational investment across 100 nodes (see Section C.5) produces a 20 TB dataset containing 12,000 samples—50% more than existing automotive datasets. The comprehensive data covers pressure distributions, velocity vectors, turbulent kinetic energy, and high-precision aerodynamic coefficients with drag coefficient accuracy of ± 0.012 . Following Loughborough University Wind Tunnel Lab specifications (Varney et al., 2020) and automobile manufacturer simulation standards, we achieve pressure coefficient errors below 1.04% compared to experimental data. Table 1 presents detailed comparisons with existing automotive CFD datasets.

To validate effectiveness, we benchmark multiple machine learning architectures, including Transolver (Wu et al., 2024), GNOT (Hao et al., 2023), and PointNet (Qi et al., 2017). Our evaluation analyzes surface pressure distributions and derived aerodynamic coefficients through surface pressure integration, establishing a multi-scale assessment for complete aerodynamic performance characterization. These benchmarks demonstrate DrivAerStar’s utility for training industrial-grade surrogate models while providing foundations for optimization research.

Our contributions are:

- **High-fidelity dataset:** DrivAerStar provides 20 TB of external flow field data from 12,000 industrial-grade STAR-CCM+[®] simulations across multiple vehicle configurations, delivering unprecedented data quality for surrogate model training.
- **Adaptive mesh methodology:** We employed refined mesh strategies with adaptive regional refinement and strict wall y^+ control, achieving superior boundary layer resolution that overcomes limitations of previous approaches.
- **Complete vehicle modeling:** Integration of front compartment assemblies with engines and cooling components creates continuous cooling flow channels, accurately representing real-world vehicle operating conditions.
- **Comprehensive benchmark:** We established multi-scale performance evaluation across surface and volumetric data, revealing current limitations and identifying research directions including scientific function discovery (Tang et al., 2023), forward Partial Differential Equation (PDE) problems (Lu et al., 2021; Raissi et al., 2019), simulation super-resolution (Kochkov et al., 2021), and inverse PDE reconstruction (Gao et al., 2021).

2 Related Work

Machine Learning CFD Benchmarks Large-scale, high-fidelity datasets form the foundation for deep learning applications in CFD. Recent standardized benchmarks support research in physical simulation and PDE modeling, including BubbleML (Hassan et al., 2023) for multiphase flows, Lagrangebench (Toshev et al., 2023) for Lagrangian particle-based simulations, BLASTNet (Chung et al., 2022) and BLASTNet 2.0 (Chung et al., 2023) for turbulent flow datasets, and PDEBench (Takamoto et al., 2022) for comprehensive PDE-based physical systems. While these datasets provide valuable contributions, they predominantly feature simplified geometries and idealized conditions that fail to capture the complex geometries and multiphysics interactions encountered in industrial applications.

Aerospace Aerodynamics Datasets Aerospace applications have driven advances in parameterized aerodynamic optimization through datasets enabling efficient simulation of complex geometries. AirfRANS (Bonnet et al., 2022) provides a 2D airfoil database built using OpenFOAM[®] with parametric meshing strategies for systematic NACA airfoil studies across operational ranges (Mach numbers 0.2–0.8, angles of attack from -8° to 20°), establishing clear mappings between geometric deformations and aerodynamic performance while providing insights into transonic shock effects. Extending to three dimensions, AircraftVerse (Cobb et al., 2023) offers 27,714 complete aircraft configurations with varied wing parameters, powertrain characteristics, and performance metrics, combining deep learning for geometry generation with engineering models for aerodynamic calculations. However, these aerospace datasets address fundamentally different flow regimes and geometric complexities than automotive applications, limiting their direct applicability to ground vehicle aerodynamics.

Automotive Aerodynamics Datasets Early automotive datasets based on simplified geometries—ShapeNet Car (Umetani and Bickel, 2018; Song et al., 2023), WindSor Car (Ashton et al., 2024a),

Table 1: **Comparative analysis of state-of-the-art automotive CFD datasets.** Quantitative benchmarking across experimental validation capabilities, numerical resolution parameters, and simulation accuracy metrics. PIV indicates flow field validation availability; wall y^+ represents dimensionless wall distance for boundary layer resolution quality; average C_D precision measures uncertainty bounds of drag coefficients; wind tunnel error shows percentage deviation from experimental measurements. Data are sourced from original literature and publicly released datasets, with detailed information available in Section E.

Metric	DrivAerNet++ (Elrefaie et al., 2024)	DrivAerML (Ashton et al., 2024c)	DrivAerStar
PIV	No	No	Yes
Engine compartment	No	No	Yes
Cooler	No	No	Yes
Samples	8,000	500	12,000
Wall y^+ range	[50, 300]	-	[30, 200]
Average C_D precision	0.012	0.010	0.005
Wind tunnel error (%)	≤ 5.0	≤ 6.5	1.04

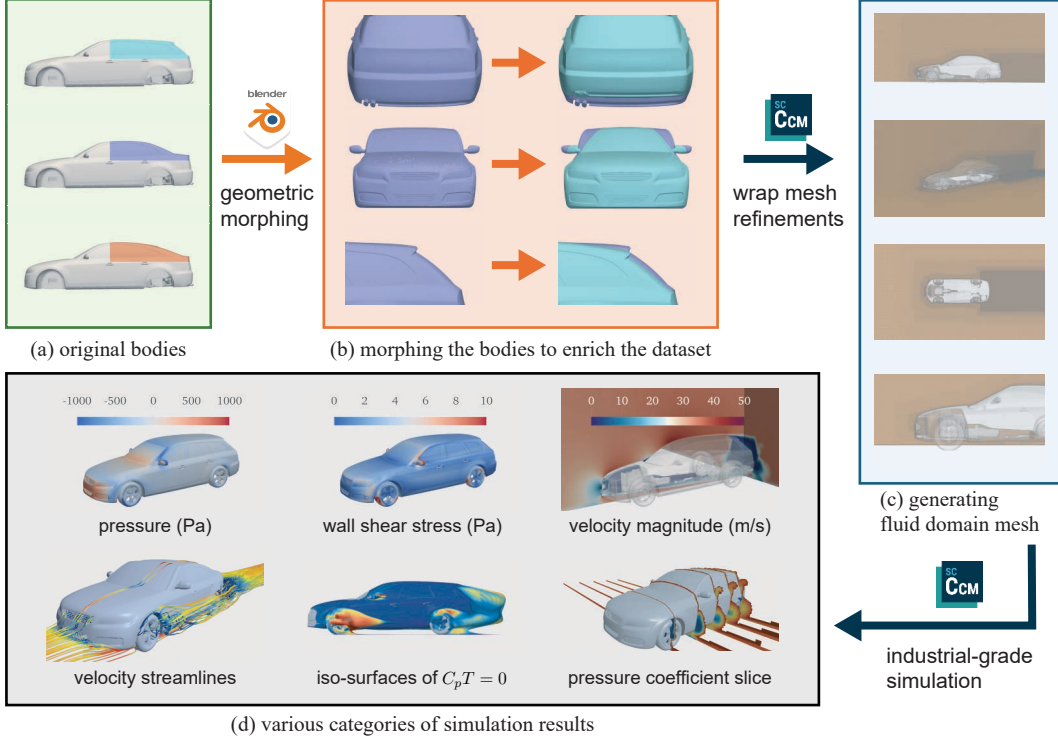


Figure 2: DrivAerStar **data generation**. (a) Three canonical DrivAer reference bodies (Estateback, Notchback, and Fastback) serve as geometric foundations. (b) Parametric morphing systematically varies 20 vehicle components, including greenhouse (top), rear diffuser (middle), and trunk lid (bottom). (c) Industrial-grade mesh generation in STAR-CCM+[®] produces refined hexahedral-dominant meshes with precise wheel alignment and boundary layer resolution for complex flow capture. (d) Comprehensive CFD simulations generate diverse flow visualizations: surface pressure, wall shear stress, velocity magnitude, streamline patterns, flow separation regions ($C_P = 0$ iso-surfaces), and pressure coefficient slices revealing key aerodynamic structures.

and Ahmed-body configurations (Li et al., 2023; Ashton et al., 2024b)—advanced the field but created substantial gaps between research outcomes and industrial requirements. More sophisticated approaches emerged with the DrivAerNet series, including DrivAerNet (Elrefaie et al., 2025) and DrivAerNet++ (Elrefaie et al., 2024), implementing parametric simulations using OpenFOAM[®]. DrivAerML similarly offers 500 parametrically morphed variants of the DrivAer Notchback model with pressure, velocity, and turbulence fields. Concurrent work by Warner and Emory (2025) introduced AeroSUV, featuring 1,000 detached eddy simulation configurations from an open-access reference model extending the DrivAer platform, executed using proprietary in-house solvers. Despite these advancements, existing automotive datasets exhibit critical limitations preventing industrial adoption: insufficient mesh resolution for complex geometries, inadequate turbulence modeling within the OpenFOAM[®] framework, and failure to account for crucial interactions between engine compartment thermal management and external aerodynamics.

DrivAerStar addresses these fundamental limitations by employing DrivAer vehicle geometry with the industry gold standard STAR-CCM+[®] solver for mesh generation and flow solution. As detailed in Table 1, our approach delivers substantial advantages in mesh resolution, aerodynamic accuracy, parametric dimensionality, and engine compartment simulation fidelity. These advancements establish a new benchmark for data-driven automotive aerodynamics research, bridging the gap between academic machine learning exploration and practical engineering applications while supporting development of general pretrained aerodynamics operator networks with potential long-term impacts detailed in Section B.

3 The DrivAerStar Dataset

Developed using industry-standard STAR-CCM+[®], DrivAerStar bridges academic research and industrial applications through rigorous validation, achieving 1.04% mean relative error compared

to wind tunnel measurements (Table 2). The dataset comprises 12,000 high-fidelity simulations of engine-integrated vehicle geometries, each containing STL representations with at least 4 million triangles and complementary data formats capturing surface distributions, full-field flow, and cross-sectional aerodynamic data. Our workflow (Figure 2) encompasses geometric morphing (Section 3.1), mesh generation (Section 3.2), and industrial-grade simulation (Section 3.3).

Unlike existing automotive CFD datasets, DrivAerStar incorporates the highest-fidelity geometric data with three rear body configurations, each featuring complete air intake, engine compartment, and cooling systems (Figure 1). To ensure industrial relevance, we systematically varied 20 fine-tuned CAD parameters (Table A1) covering both local and global vehicle features using Blender® through Latin Hypercube Sampling (LHS), creating a comprehensive design exploration space (Figure A2). The dataset spans Reynolds numbers from 9.46×10^6 to 1.48×10^7 , with flow features available in volumetric, slice, and surface formats including velocity, pressure, surface properties, wall shear stress, mesh configurations, y^+ values, 3D streamlines, and iso-surfaces.

DrivAerStar provides comprehensive aerodynamic metrics aligned with physics field data: pressure coefficient (C_P), friction coefficient (C_F), drag coefficient (C_D), and lift coefficient (C_L). Beyond performance metrics, we supply complete numerical data (pressure, velocity, wall shear stress) alongside reproducible workflows with validation scripts following industrial standards. Simulation results are detailed in Section C.6. The dataset subset, encompassing over 12,000 VTK surface files and EnSight flow field files totaling approximately 20 TB, is accessible via our project homepage under the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0) license, with data processing and benchmarking code included. Detailed licensing information is provided in Section A.

3.1 Geometric Morphing

DrivAerStar builds upon the experimentally validated DrivAer reference vehicle platform (Heft et al., 2012; Strangfeld et al., 2013), featuring three canonical rear configurations: Fastback, Estateback, and Notchback. To generate our comprehensive dataset of 12,000 geometrically diverse vehicle models, we implemented systematic FFD techniques targeting critical aerodynamic components. This approach (Figure 3) enables precise parametric control over vehicle geometry, transforming the base Fastback model into varied body and wheel configurations through controlled parameter adjustments. Complete technical details of our FFD implementation are provided in Section C.1.

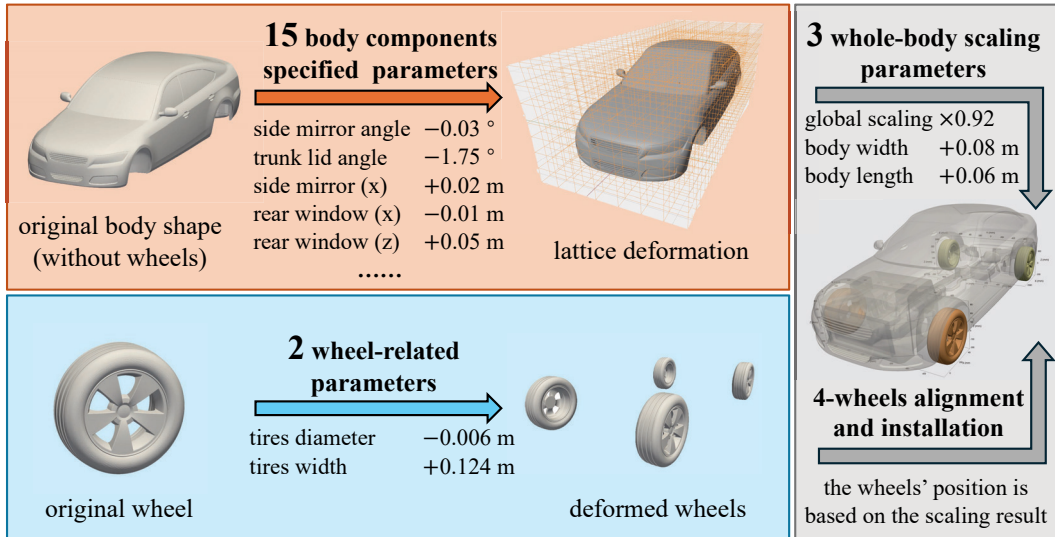


Figure 3: **Geometric morphing pipeline example.** Our parametric deformation framework transforms the baseline model through three sequential operations: (i) **Body component morphing** (top-left) applies 15 controlled parameter adjustments, including dimensional modifications and component repositioning; (ii) **Wheel morphing** (bottom-left) enables precise tire parameter control; and (iii) **Whole-body scaling and wheel installation** (right) applies 3 scaling parameters while aligning wheels to scaled positions. This systematic FFD approach generates geometrically diverse yet aerodynamically realistic vehicle configurations.

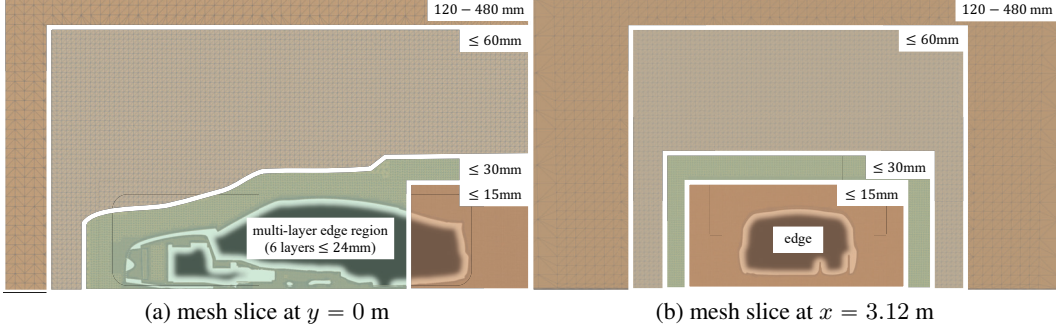


Figure 4: **Cross-sectional views of mesh generation strategy.** (a) Longitudinal section at $y = 0$ m shows four nested refinement zones with progressive cell density: far-field (120-480 mm), intermediate domain (≤ 60 mm), near-body region (≤ 30 mm), and high-resolution zone (≤ 15 mm) surrounding the vehicle. Dedicated 6-layer boundary layer refinement with total thickness under 24 mm captures viscous phenomena. (b) Transverse section at $x = 3.12$ m demonstrates consistent refinement strategy with graduated cell density approaching vehicle surfaces, enabling accurate wake structure resolution while maintaining computational efficiency.

3.2 Mesh Generation

Our automated mesh generation workflow consists of two sequential stages: surface wrapping and volume mesh generation.

Surface Wrapping Morphed geometries frequently exhibit topological defects, including mesh intersections and open edges, requiring systematic resolution. We employed STAR-CCM+[®]'s surface wrapping functionality on individual components—vehicle body, drivetrain, engine compartment, wheels, and coolers—to ensure simulation-ready geometry. Critical regions with fine features, particularly drivetrain and grille openings, received 10% relative size refinement and specialized contact prevention parameters to maintain geometric fidelity while eliminating surface intersections. Each component underwent independent wrapping before integration into the computational domain. Complete technical specifications are detailed in [Section C.2](#).

Volume Mesh Generation Our volume mesh ([Figure 4](#)) employs refinement strategies aligned with validated practices ([Heft et al., 2012](#); [Zhang et al., 2019](#)) using STAR-CCM+[®]'s production-grade algorithms. Each simulation domain comprises approximately 12 million hexahedral-dominant cells with strategic density distribution targeting aerodynamically significant features. The meshing protocol incorporates high-resolution prismatic boundary layers on all vehicle surfaces—both external and internal—ensuring proper resolution of boundary phenomena including separation, reattachment, and near-wall flow structures. To maintain dataset consistency, we automated the entire meshing sequence through Java macros with identical parameters across all 12,000 simulations. Additional details are provided in [Section C.3](#).

3.3 Industrial-grade Simulation

Computational Domain and Boundary Conditions Our framework implements automotive industry-standard simulation protocols using a domain extending 15 vehicle lengths streamwise, 10 widths laterally, and 12 heights vertically. The inlet boundary, positioned three vehicle lengths upstream, imposes 40 m/s (144 km/h) freestream velocity with 1% turbulence intensity generated through spectral synthesizer algorithms. The downstream boundary enforces zero-gradient pressure outlet conditions. Vehicle-ground interaction employs sliding wall boundaries with rotating reference frames, calculating wheel angular velocity ($\omega = u_\infty/r$) based on instantaneous tire geometry. The engine compartment thermal management system incorporates experimentally validated porous media models with 0.6 porosity. All simulations utilize T/CSAE112-2019 standard atmospheric conditions: air density $\rho = 1.225$ kg/m³ and dynamic viscosity $\mu = 1.85508 \times 10^{-5}$ Pa · s.

Flow Physics Modeling Simulations solve 3D steady-state Reynolds-Averaged Navier–Stokes (RANS) equations ([Reynolds, 1895](#)) for incompressible flow with comprehensive wall distance modeling. We employed Shear-Stress Transport (SST) ([Menter, 1994](#)) k - ω turbulence formulation, selected for demonstrated accuracy in predicting flow separation and reattachment phenomena

critical to automotive aerodynamics. Our hybrid all- y^+ wall treatment automatically transitions between direct viscous sublayer resolution and wall function approaches based on local mesh refinement, maintaining y^+ values below 200 while optimizing computational efficiency. Gradient reconstruction utilizes cell-based least squares minimization, ensuring numerical stability in regions with strong pressure gradients and separated flow. Complete governing equation formulations are provided in [Section C.4](#).

Sovler Our solution strategy employs a pressure-based segregated solver with Semi-Implicit Method for Pressure-Linked Equations (SIMPLE) algorithm (Patankar and Spalding, 1983) for pressure-velocity coupling. Spatial discretization implements second-order schemes for pressure and momentum equations, with first-order upwind treatment for turbulence quantities, enhancing stability. Linear equation systems are resolved using Algebraic Multigrid (AMG) methods (Xu and Zikatanov, 2017) with V-cycle iterations (1 pre/post-sweep), achieving convergence to residuals below 10^{-5} within 30 iterations per time step. Carefully calibrated under-relaxation factors (0.7 for velocity, 0.8 for turbulence quantities) balance convergence rate with solution stability. Gauss-Seidel relaxation enhances iterative efficiency, particularly in high-gradient regions such as wheel wells and underbody flow channels.

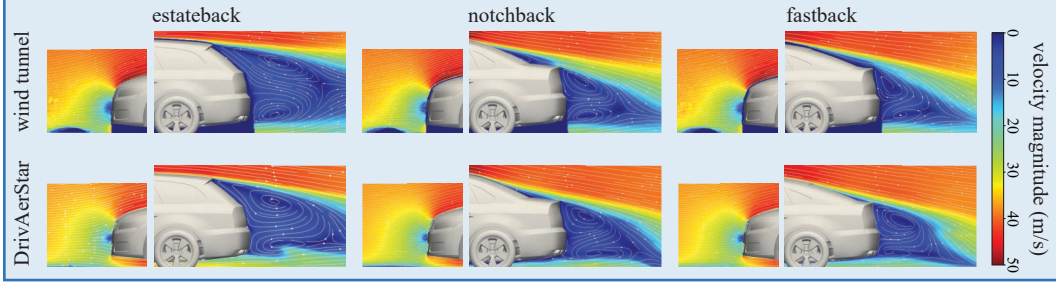
3.4 Experimental Validation

Wind Tunnel Validation We validated our simulation methodology against high-fidelity measurements from Loughborough University Large Wind Tunnel facility (Johl et al., 2004), focusing on drag coefficient (C_D)—the critical automotive aerodynamics performance metric. Our 25%-scale simulations precisely matched physical models provided by FKFS Stuttgart, ensuring geometric consistency between computational and experimental configurations. As shown in [Table 2](#), simulations demonstrate exceptional agreement with experimental measurements across all three rear body configurations (Fastback, Notchback, Estateback), achieving 1.04% average relative deviation with maximum discrepancy below 1.8%. This correlation substantiates our simulation methodology’s predictive accuracy for aerodynamic performance assessment. Detailed convergence analysis and mesh independence studies are presented in [Section C.7](#).

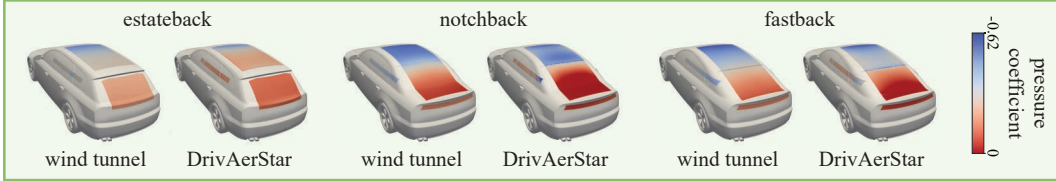
Table 2: **Data scaling effects on drag coefficient prediction accuracy.** Transolver model performance analysis with increasing training sample size. Left and center: Scatter plots of predicted versus ground truth C_D values for models trained on 400 and 1200 samples, respectively, with diagonal green line representing perfect prediction ($y = x$) and red regression line showing actual prediction trends with correlation coefficients. Right: Violin plots comparing prediction error distributions, demonstrating improved performance with increased data volume (mean absolute error decreasing from 2.99% to 2.78%). Results confirm `DrivAerStar`’s effective learning scaling properties and potential for further accuracy improvements with additional training data. Comprehensive scaling results across all configurations are detailed in [Section D.2](#).

Rear Configuration	Wind Tunnel C_D	DrivAerStar C_D	Deviation (%)
FastBack	0.278	0.2749	1.12%
NotchBack	0.279	0.2767	0.82%
Estateback	0.299	0.2955	1.17%

PIV Validation Flow structure prediction was further validated through Particle Image Velocimetry (PIV) measurements in aerodynamically critical regions—front stagnation zones, A-pillar vortices, and rear wake structures. Surface pressure distributions were compared using data from 60 strategically positioned pressure taps, emphasizing the interchangeable rear configurations. As visualized in [Figure 5](#), these comparisons demonstrate excellent agreement in both magnitude and spatial distribution of pressure coefficients and velocity fields. The observed congruence confirms that our dataset accurately captures complex aerodynamic phenomena governing vehicle performance, including internal engine bay flows typically omitted in simplified models. Quantitative simulation-experiment difference analysis is provided in [Section C.8](#).



(a) wake and stagnation regime by PIV images



(b) time-averaged C_P

Figure 5: **Validation of flow physics predictions against wind tunnel measurements.** (a) Velocity magnitude distributions compare wind tunnel PIV measurements (top row) with DrivAerStar predictions (bottom row) in wake regions and stagnation zones, demonstrating excellent flow structure agreement. (b) Surface pressure coefficient (C_P) distributions across vehicle body surfaces show consistent prediction accuracy between experimental and computational results for all three vehicle configurations, confirming accurate capture of complex aerodynamic phenomena.

4 Benchmarks on DrivAerStar

4.1 Experimental Setup

Benchmark Objectives We establish comprehensive machine learning benchmarks on the DrivAerStar dataset to evaluate its industrial relevance and predictive capability. These benchmarks assess state-of-the-art deep learning architectures on aerodynamic prediction tasks using high-fidelity CFD data representative of real automotive development. The primary task involves predicting surface pressure (\bar{p}) and wall shear stress ($\bar{\tau}_w$) distributions from vehicle geometry, as well as the derived drag coefficients (C_D). This setup aligns with recent advances in automotive aerodynamics machine learning research (Hao et al., 2023; Wu et al., 2024; Nabian et al., 2024; Choy et al., 2025; Liu et al., 2025; Bleeker et al., 2025) while providing realistic validation through our industry-standard dataset.

Dataset Configurations We systematically evaluated dataset scale and diversity effects. Initial benchmarks used restricted training sets (400-1200 samples) from single vehicle configurations (Estateback), while advanced experiments incorporated multi-configuration datasets spanning all three rear body types (Fastback, Estateback, Notchback). All experiments maintained consistent validation and test splits with 150 cases per configuration type. Additional experimental details are provided in Section F.

Evaluation Protocol Model performance evaluation ensures accurate local flow physics representation and reliable global performance prediction—essential requirements for industrial applications. For spatial distribution accuracy of surface fields, we employ the relative L_2 error. Each flow field variable ϕ (e.g., velocity, pressure) is standardized via mean (μ_ϕ) and standard deviation (σ_ϕ) computed from dataset samples (i.e., $\hat{\phi} = \frac{\phi - \mu_\phi}{\sigma_\phi}$): $\epsilon = \frac{\|\hat{\phi}_{\text{pred}} - \hat{\phi}_{\text{true}}\|_2}{\|\hat{\phi}_{\text{true}}\|_2}$, eliminating scale discrepancies among variables, where $\hat{\phi}_{\text{pred}}$ and $\hat{\phi}_{\text{true}}$ denote standardized predicted and ground-truth flow field variables, respectively. For integrated quantities (e.g., drag coefficient C_D), we assess absolute and percentage errors to ensure correct global performance prediction. Complete metric definitions and standardization procedures are provided in Section D.1.

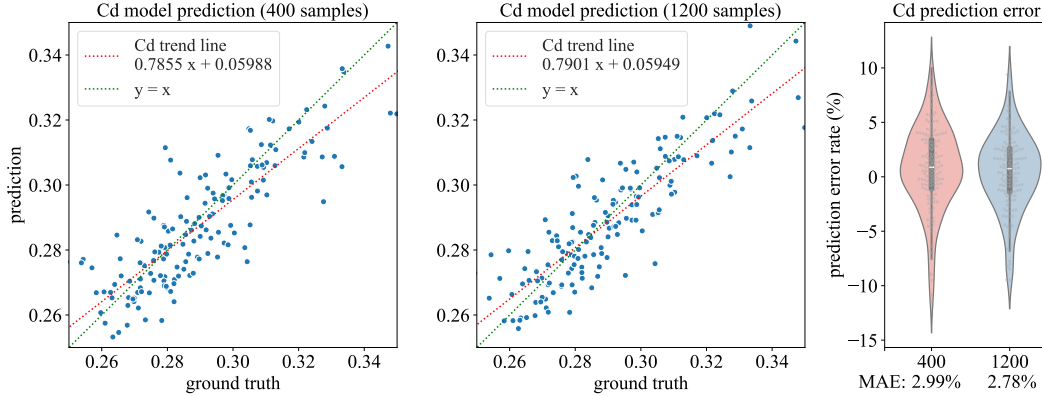


Figure 6: **Data scaling effects on drag coefficient prediction accuracy.** Analysis of Transolver model performance with increasing training sample size. Left and center: Scatter plots of predicted versus ground truth C_D values for models trained on 400 and 1200 samples, respectively. The diagonal green line represents perfect prediction ($y = x$), while the red regression line shows the actual prediction trend with correlation coefficients. Right: Violin plots comparing prediction error distributions, showing improved performance with increased data volume (mean absolute error decreasing from 2.99% to 2.78%). This demonstrates the dataset’s effective learning scaling properties and potential for further accuracy improvements with additional samples. Comprehensive scaling results across all configurations are provided in [Section D.2](#).

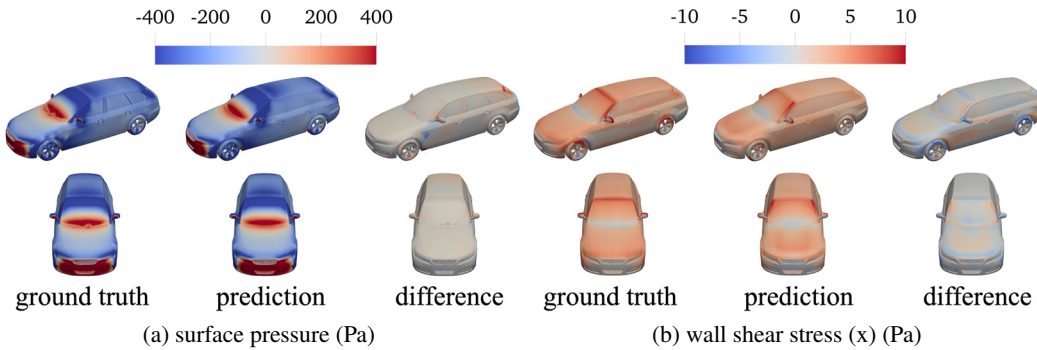


Figure 7: **Surface field prediction accuracy across critical aerodynamic regions.** (a) Surface pressure distributions (Pa) comparing ground truth CFD simulation (left), neural network prediction (middle), and point-wise difference (right). (b) Wall shear stress τ_x (Pa) in streamwise direction with ground truth (left), model prediction (middle), and prediction error (right). The model accurately captures high-gradient regions around A-pillars and wheel wells while maintaining precision in uniform flow areas, demonstrating robust performance across diverse aerodynamic phenomena.

4.2 Results

Surface Field Prediction We evaluated three state-of-the-art deep learning architectures on *DriveAerStar* for predicting surface pressure and wall shear stress fields, as well as the derived drag coefficients. The performance metrics on the test set are summarized in [Table A2](#).

Each model was trained on both specialized single-configuration subsets (Estateback only) and the complete multi-configuration dataset (Fastback, Notchback, and Estateback). Results consistently showed higher prediction errors on the complete dataset, confirming that morphological diversity significantly increases prediction complexity. This performance differential quantifies the fundamental trade-off between prediction accuracy and geometric variability in aerodynamic machine learning tasks. Representative prediction visualizations are presented in [Figure 7](#).

Data Scaling Analysis To investigate model generalization capabilities and data efficiency, we conducted systematic scaling experiments following *DriveAer* protocols. Both single-configuration subsets and complete datasets were divided into training (80%) and test (20%) partitions, with progressive increases in training samples while maintaining consistent test evaluation. The resulting learning curves ([Figure 6](#)) reveal characteristic scaling behaviors for each architecture and quantify data requirements for achieving specific prediction accuracy thresholds.

Table 3: **Drag coefficient prediction scaling analysis in multi-vehicle configurations.** Transolver model performance across varying training dataset sizes (400–12,000 samples) evaluated on a consistent validation set of 150 samples per vehicle configuration. Learning curves demonstrate systematic improvement in C_D prediction accuracy with increased training data, confirming the dataset’s scalability and the model’s ability to leverage additional samples for enhanced aerodynamic performance prediction across diverse vehicle geometries.

Metrics	400	800	1,200	12,000
C_D validation loss	0.0375	0.0335	0.0286	0.0266
Improvement	-	10.67%	23.73%	29.07%

We further assessed training size effects on Transolver’s predictive performance for drag coefficient (C_D)—a key CFD metric for vehicle aerodynamic optimization—in multi-vehicle settings. Results including C_D validation loss and relative improvement values are summarized in Table 3. Raw scaling experiment data (loss curves, test sample distributions) are provided in Section D.3.

5 Conclusion

We present *DrivAerStar*, a comprehensive automotive aerodynamics dataset comprising 12,000 high-fidelity CFD simulations that bridges the gap between academic research and industrial applications. Generated using industry-standard STAR-CCM+[®] software across three rear body configurations with 20 systematically varied CAD parameters, *DrivAerStar* represents the most comprehensive and validated automotive aerodynamics dataset to date.

Our key contributions establish new standards for CFD datasets in automotive engineering. We provide the highest-fidelity geometric representations available, incorporating complete engine compartments, cooling systems, and drivetrain components typically omitted in simplified academic models. The systematic parametric morphing using FFD techniques creates geometric diversity while maintaining industrial relevance. Our rigorous experimental validation against wind tunnel measurements achieves exceptional accuracy with 1.04% mean relative error for drag coefficients and excellent agreement in PIV flow field comparisons, establishing confidence in the dataset’s predictive capability for real-world applications.

The industrial-grade simulation methodology employs production-level meshing protocols with 12 million hexahedral-dominant cells per case, capturing boundary layer phenomena and separation dynamics critical to automotive aerodynamics. Our comprehensive data formats provide complete access to flow physics information. The resulting 20TB dataset includes pressure coefficients, wall shear stress, velocity fields, streamlines, and iso-surfaces, enabling diverse research applications from fundamental flow physics to applied design optimization.

Our extensive machine learning benchmarks demonstrate that models trained on *DrivAerStar* achieve industrial-grade prediction accuracy while reducing computational costs by orders of magnitude compared to traditional CFD approaches. The systematic evaluation reveals fundamental trade-offs between geometric variability and prediction accuracy, providing crucial insights for developing robust aerodynamic prediction models and guiding future dataset development efforts.

DrivAerStar’s impact extends beyond immediate research applications to enable transformative changes in automotive development processes. By providing validated, high-fidelity training data, we enable the development of fast, accurate surrogate models that can replace computationally expensive CFD simulations in design optimization loops. This supports rapid design exploration, real-time performance assessment, and integration of aerodynamic considerations throughout the vehicle development cycle.

The open availability of *DrivAerStar* under permissive licensing, combined with comprehensive documentation and reproducible workflows, establishes a foundation for collaborative research in automotive aerodynamics. Future extensions will address current limitations, including multidisciplinary coupling with thermal management and structural dynamics, detailed in Section G.

By democratizing access to industrial-quality aerodynamic data, *DrivAerStar* empowers researchers worldwide to develop next-generation computational methods that accelerate the transition to more efficient, sustainable automotive designs. This work represents a critical step toward data-driven automotive engineering, where machine learning and high-fidelity simulation combine to revolutionize vehicle development practices.

Acknowledgments and Disclosure of Funding

Deep learning experiments were conducted using the PaddlePaddle framework. We gratefully acknowledge Siemens Digital Industries Software for granting an educational-discount license of STAR-CCM+® to Peking University, enabling part of the CFD simulations presented in this study. Y. Xu, L. Cui, and Y. Zhu were supported by the National Natural Science Foundation of China (62376009), the PKU-BingJi Joint Laboratory for Artificial Intelligence, and the National Comprehensive Experimental Base for Governance of Intelligent Society, Wuhan East Lake High-Tech Development Zone.

References

- Abinesh, J. and Arunkumar, J. (2014). Cfd analysis of aerodynamic drag reduction and improve fuel economy. *International Journal of Mechanical Engineering and Robotics Research*, 3(4):430. [1](#)
- Ashton, N., Angel, J., Ghate, A., Kenway, G., Wong, M. L., Kiris, C., Walle, A., Maddix, D., and Page, G. (2024a). Windsorml: High-fidelity computational fluid dynamics dataset for automotive aerodynamics. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*. [3](#)
- Ashton, N., Maddix, D. C., Gundry, S., and Shabestari, P. M. (2024b). Ahmedml: High-fidelity computational fluid dynamics dataset for incompressible, low-speed bluff body aerodynamics. [4](#)
- Ashton, N., Mockett, C., Fuchs, M., Fliessbach, L., Hetmann, H., Knacke, T., Schonwald, N., Skaperdas, V., Fotiadis, G., Walle, A., et al. (2024c). Drivaerml: High-fidelity computational fluid dynamics dataset for road-car external aerodynamics. [2](#), [3](#), [A12](#)
- Bibra, E. M., Connelly, E., Dhir, S., Drtil, M., Henriot, P., Hwang, I., Le Marois, J.-B., McBain, S., Paoli, L., and Teter, J. (2022). Global ev outlook 2022: Securing supplies for an electric future. Technical report, IEA. [1](#)
- Bjerkkan, K. Y., Nørbech, T. E., and Nordtømme, M. E. (2016). Incentives for promoting battery electric vehicle (bev) adoption in norway. *Transportation Research Part D: Transport and Environment*, 43:169–180. [2](#)
- Bleeker, M., Dorfer, M., Kronlachner, T., Sonnleitner, R., Alkin, B., and Brandstetter, J. (2025). Neuralcfd: Deep learning on high-fidelity automotive aerodynamics simulations. [8](#)
- Bonnet, F., Mazari, J., Cinnella, P., and Gallinari, P. (2022). Airfrans: High fidelity computational fluid dynamics dataset for approximating reynolds-averaged navier–stokes solutions. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*. [3](#)
- Brandt, A., Jacobson, B., and Sebben, S. (2022). High speed driving stability of road vehicles under crosswinds: an aerodynamic and vehicle dynamic parametric sensitivity analysis. *Vehicle System Dynamics*, 60(7):2334–2357. [1](#)
- Choy, C., Kamenev, A., Kossaifi, J., Rietmann, M., Kautz, J., and Azizzadenesheli, K. (2025). Factorized implicit global convolution for automotive computational fluid dynamics prediction. [8](#)
- Chung, W. T., Akoush, B., Sharma, P., Tamkin, A., Jung, K. S., Chen, J., Guo, J., Brouzet, D., Talei, M., Savard, B., et al. (2023). Turbulence in focus: Benchmarking scaling behavior of 3d volumetric super-resolution with blastnet 2.0 data. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*. [3](#)
- Chung, W. T., Jung, K. S., Chen, J. H., and Ihme, M. (2022). Blastnet: A call for community-involved big data in combustion machine learning. *Applications in Energy and Combustion Science*, 12:100087. [3](#)
- Cobb, A., Roy, A., Elenius, D., Heim, F., Swenson, B., Whittington, S., Walker, J., Bapty, T., Hite, J., Ramani, K., et al. (2023). Aircraftverse: a large-scale multimodal dataset of aerial vehicle designs. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*. [3](#)

- Elrefaie, M., Dai, A., and Ahmed, F. (2025). Drivaernet: A parametric car dataset for data-driven aerodynamic design and prediction. *Journal of Mechanical Design*, 147(4). 4
- Elrefaie, M., Morar, F., Dai, A., and Ahmed, F. (2024). Drivaernet++: A large-scale multimodal car dataset with computational fluid dynamics simulations and deep learning benchmarks. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*. 2, 3, 4, A12
- Gao, Q., Pan, S., Wang, H., Wei, R., and Wang, J. (2021). Particle reconstruction of volumetric particle image velocimetry with the strategy of machine learning. *Advances in Aerodynamics*, 3(1):28. 3
- Hannan, M. A., Azidin, F., and Mohamed, A. (2014). Hybrid electric vehicles and their challenges: A review. *Renewable and Sustainable Energy Reviews*, 29:135–150. 2
- Hao, Z., Wang, Z., Su, H., Ying, C., Dong, Y., Liu, S., Cheng, Z., Song, J., and Zhu, J. (2023). Gnot: A general neural operator transformer for operator learning. In *Proceedings of International Conference on Machine Learning (ICML)*. 2, 8, A13
- Hassan, S. M. S., Feeney, A., Dhruv, A., Kim, J., Suh, Y., Ryu, J., Won, Y., and Chandramowlishwaran, A. (2023). Bubbleml: A multiphase multiphysics dataset and benchmarks for machine learning. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*. 3
- Heft, A. I., Indinger, T., and Adams, N. A. (2012). Experimental and numerical investigation of the driver model. In *Fluids Engineering Division Summer Meeting*. 5, 6
- Huluka, A. and Kim, C. (2019). Numerical study on aerodynamic drag reduction and energy harvest for electric vehicle: a concept to extend driving range. In *IOP Conference Series: Materials Science and Engineering*. 1
- Johl, G., Passmore, M., and Render, P. (2004). Design methodology and performance of an indraft wind tunnel. *The Aeronautical Journal*, 108(1087):465–473. 7
- Kochkov, D., Smith, J. A., Alieva, A., Wang, Q., Brenner, M. P., and Hoyer, S. (2021). Machine learning–accelerated computational fluid dynamics. *Proceedings of the National Academy of Sciences (PNAS)*, 118(21):e2101784118. 3
- Li, Z., Kovachki, N. B., Choy, C., Li, B., Kossaiji, J., Otta, S. P., Nabian, M. A., Stadler, M., Hundt, C., Azizzadenesheli, K., and Anandkumar, A. (2023). Geometry-informed neural operator for large-scale 3d pdes. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*. 4
- Liu, P., Wang, P., Ren, X., Yuan, H., Hao, Z., Xu, C., Cai, S., and Ni, D. (2025). Aerocto: An efficient graph-transformer operator for learning large-scale aerodynamics of 3d vehicle geometries. In *Proceedings of AAAI Conference on Artificial Intelligence (AAAI)*. 8
- Lu, L., Jin, P., Pang, G., Zhang, Z., and Karniadakis, G. E. (2021). Learning nonlinear operators via deeponet based on the universal approximation theorem of operators. *Nature machine intelligence*, 3(3):218–229. 3
- Menter, F. R. (1994). Two-equation eddy-viscosity turbulence models for engineering applications. *AIAA Journal*, 32(8):1598–1605. 6, A7
- Nabian, M. A., Liu, C., Ranade, R., and Choudhry, S. (2024). X-meshgraphnet: Scalable multi-scale graph neural networks for physics simulation. 8
- Patankar, S. V. and Spalding, D. B. (1983). A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows. In *Numerical prediction of flow, heat transfer, turbulence and combustion*, pages 54–73. Elsevier. 7
- Payri, F., Luján, J. M., Guardiola, C., and Pla, B. (2015). A challenging future for the ic engine: new technologies and the control role. *Oil & Gas Science and Technology–Revue d’IFP Energies nouvelles*, 70(1):15–30. 2

- Qi, C. R., Su, H., Mo, K., and Guibas, L. J. (2017). Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR)*. 2, A12
- Raissi, M., Perdikaris, P., and Karniadakis, G. E. (2019). 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. 3
- Reynolds, O. (1895). On the dynamical theory of incompressible viscous fluids and the determination of the criterion. *Philosophical Transactions of the Royal Society of London (A)*, 186:123–164. 6
- Schlichting, H. and Kestin, J. (1961). *Boundary layer theory*, volume 121. Springer. 2
- Song, B., Yuan, C., Permenter, F., Arechiga, N., and Ahmed, F. (2023). Surrogate modeling of car drag coefficient with depth and normal renderings. In *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*. 3
- Spalart, P. R. and Rumsey, C. L. (2007). Effective inflow conditions for turbulence models in aerodynamic calculations. *AIAA journal*, 45(10):2544–2553. A7
- Strangfeld, C., Wieser, D., Schmidt, H.-J., Woszidlo, R., Nayeri, C., and Paschereit, C. (2013). Experimental study of baseline flow characteristics for the realistic car model drivaer. Technical report, SAE Technical Paper. 5
- Stüben, K. (2001). A review of algebraic multigrid. *Numerical Analysis: Historical Developments in the 20th Century*, pages 331–359. A8
- Takamoto, M., Praditia, T., Leiteritz, R., MacKinlay, D., Alesiani, F., Pflüger, D., and Niepert, M. (2022). Pdebench: An extensive benchmark for scientific machine learning. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*. 3
- Tang, H., Wang, Y., Wang, T., and Tian, L. (2023). Discovering explicit reynolds-averaged turbulence closures for turbulent separated flows through deep learning-based symbolic regression with non-linear corrections. *Physics of Fluids*, 35(2). 3
- Toshev, A., Galletti, G., Fritz, F., Adami, S., and Adams, N. (2023). Lagrangebench: A lagrangian fluid mechanics benchmarking suite. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*. 3
- Umetani, N. and Bickel, B. (2018). Learning three-dimensional flow for interactive aerodynamic design. *ACM Transactions on Graphics (TOG)*, 37(4):1–10. 3
- Upadhyaya, A. and Mahanta, C. (2023). An overview of battery based electric vehicle technologies with emphasis on energy sources, their configuration topologies and management strategies. *IEEE Transactions on Intelligent Transportation Systems*, 25(2):1087–1111. 2
- Varney, M., Passmore, M., Wittmeier, F., and Kuthada, T. (2020). Experimental data for the validation of numerical methods: Drivaer model. *Fluids*, 5(4):236. 2
- Wang, Y., Zhen, X., Wu, J., Gu, Z., Xiao, Z., and Yang, X. (2013). Hybrid cfd/fem-bem simulation of cabin aerodynamic noise for vehicles traveling at high speed. *Science China Technological Sciences*, 56:1697–1708. 1
- Warner, J. and Emory, M. (2025). Introducing luminary shift models: A suite of physics ai foundation models to transform engineering design. Technical report, Luminary Cloud. 4
- Wu, H., Luo, H., Wang, H., Wang, J., and Long, M. (2024). Transolver: A fast transformer solver for pdes on general geometries. In *Proceedings of International Conference on Machine Learning (ICML)*. 2, 8, A12
- Xu, J. and Zikatanov, L. (2017). Algebraic multigrid methods. *Acta Numerica*, 26:591–721. 7
- Zhang, C., Bounds, C. P., Foster, L., and Uddin, M. (2019). Turbulence modeling effects on the cfd predictions of flow over a detailed full-scale sedan vehicle. *Fluids*, 4(3):148. 6

NeurIPS Paper Checklist

1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper’s contributions and scope?

Answer: [Yes]

Justification: Yes, the main claims made in the abstract and introduction accurately reflect our contribution and scope, which is the comprehensive, reproducible, and industrial-grade dataset `DriveStar`.

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [Yes]

Justification: We have discussed the limitations in [Section G](#), including insufficient geometric deformation to cover all designs in actual industrial development, and the CFD solver may still deviate from the real-industry conditions.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (*e.g.*, independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, *e.g.*, if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren’t acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. Theory assumptions and proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: We have provided a detailed explanation of the foundational theories of fluid dynamics and a thorough simulation setup clarification according to the theories, which can be found at [Section C.4](#).

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. Experimental result reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We provide the implementation details of the data generation pipeline at [Sections 3](#) and [C](#).

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (*e.g.*, in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
 - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
 - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
 - (c) If the contribution is a new model (*e.g.*, a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (*e.g.*, with an open-source dataset or instructions for how to construct the dataset).

- (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (*e.g.*, to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: Both `DrivAerStar` and code for data processing and benchmarking are accessible via our project [homepage](#). We provide the detailed instructions for our dataset and code on the related websites.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (*e.g.*, for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. Experimental setting/details

Question: Does the paper specify all the training and test details (*e.g.*, data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: We have made clear the dataset generation settings, reported how we validate and benchmark the SOTA models. More details can be found in the [Sections D.2](#) and [F](#).

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. Experiment statistical significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: Due to computational resource constraints, we were unable to establish the statistical significance of our experimental results.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (*e.g.*, Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (*e.g.* negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. Experiments compute resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: We have stated the high-performance computation cluster configuration and CPU core-hours needed in this paper to produce `DriveAerStar` in [Section C.5](#).

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (*e.g.*, preliminary or failed experiments that didn't make it into the paper).

9. Code of ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>?

Answer: [Yes]

Justification: We have adhered to the NeurIPS Code of Ethics throughout the entire process of our research.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (*e.g.*, if there is a special consideration due to laws or regulations in their jurisdiction).

10. Broader impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: We have stated that our work could bring positive societal impacts through boosting the automotive industry’s development cycle and even extending beyond the field of automotive aerodynamics in [Section B](#).

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (*e.g.*, disinformation, generating fake profiles, surveillance), fairness considerations (*e.g.*, deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (*e.g.*, gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (*e.g.*, pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: Our paper poses no risk since we have only collected data from credible sources.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. Licenses for existing assets

Question: Are the creators or original owners of assets (*e.g.*, code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: We’ve credited the owner of `DrivAer` assets and stated the licenses of software we used (STAR-CCM+[®] and Blender[®]) in the paper.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (*e.g.*, CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (*e.g.*, website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: Both `DrivAerStar` and code for data processing and benchmarking are well documented, and are accessible via our project [homepage](#).

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: This paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: This paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: LLMs are not involved in our core method development in this research.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (<https://neurips.cc/Conferences/2025/LLM>) for what should or should not be described.

A Resources Availability and Licensing

DrivAerStar is distributed under the CC BY-NC-SA 4.0 license to facilitate academic dissemination while protecting intellectual property rights. Under this license, users are permitted to copy, distribute, display, and perform the work, as well as create derivative works, provided that they give appropriate credit to the original authors, provide a link to the license, and indicate if changes were made. **Commercial use of the work is strictly prohibited without explicit written permission from the copyright holders.**

All associated resources, including source code, datasets, and documentation, are publicly accessible to ensure transparency and enable reproducibility. Users are encouraged to contribute improvements through established channels, enhancing the collective value of these open-source resources within the academic community.

B Broader Impact

DrivAerStar addresses critical needs across multiple research and industrial domains. In academic research, the high-fidelity dataset enables development of novel geometric deep learning models for complex fluid dynamics, advancing understanding of how neural networks capture physical phenomena, and supporting investigations into physics-informed machine learning approaches. The dataset expands available benchmarks for graph neural networks and point cloud methods, establishing new evaluation standards for fluid dynamics applications.

For industrial applications, DrivAerStar enables development of fast surrogate models that can replace computationally expensive CFD simulations in design optimization workflows. This capability supports more efficient automotive development cycles and extends to aerospace applications where virtual testing reduces reliance on expensive physical prototypes. The dataset addresses growing industrial demand for understanding deep learning generalization across large solution domains, promoting integration of high-fidelity physics-based simulations with artificial intelligence.

Beyond immediate applications, DrivAerStar establishes a foundation for data-driven design methodologies that can transform engineering practices across disciplines facing computational constraints. By democratizing access to industrial-quality aerodynamic data, the dataset empowers researchers worldwide to develop next-generation computational methods for more efficient and sustainable engineering designs.

Table A1: **Parametric deformation ranges for automotive geometry generation via lattice-based morphing.** Twenty parameters control geometric variations across vehicle components, with ranges established from production vehicle design standards and aerodynamic optimization requirements.

Number	Parameter	Minimum	Maximum	Unit
1	Global Scaling	80%	120%	-
2	Vehicle Width	-0.1	0.1	m
3	Vehicle Length	-0.1	0.1	m
4	Ramp Angle	-10	10	degree
5	Front Bumper Length	-0.1	0.1	m
6	Windscreen X	-0.05	0.05	m
7	Windscreen Z	-0.05	0.05	m
8	Side Mirrors X	-0.05	0.05	m
9	Side Mirrors Z	-0.05	0.05	m
10	Rear Window X	-0.05	0.05	m
11	Rear Window Z	-0.05	0.05	m
12	Trunk Lid Angle	-10	10	degree
13	Trunk Lid X	-0.05	0.05	m
14	Trunk Lid Z	-0.05	0.05	m
15	Diffuser Angle	-10	10	degree
16	Greenhouse Angle	-10	10	degree
17	Front Hood Angle	-10	10	degree
18	Intake Hood Angle	-10	10	degree
19	Tire Diameter	-0.033	0.033	m
20	Tire Width	-0.015	0.015	m

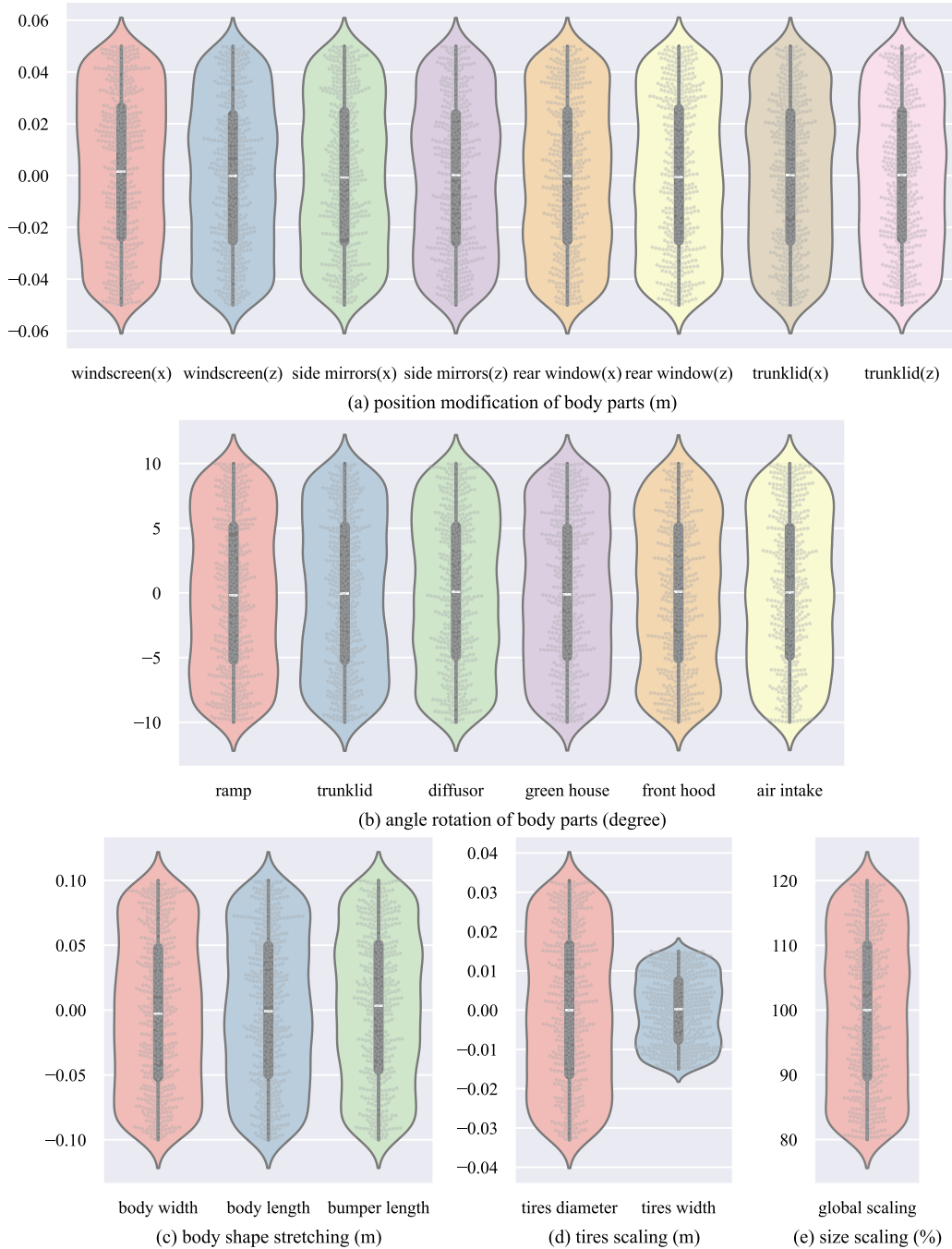


Figure A1: **Parametric deformation diversity in DriveAerStar.** Five deformation categories generate geometric variations through Latin Hypercube Sampling: (a) Linear displacement of four vehicle components within $\pm 0.05\text{m}$ in x and z directions. (b) Angular rotation of six body elements ranging from -10° to 10° . (c) Stretching and compression of vehicle body and front bumper between $\pm 0.1\text{m}$. (d) Tire deformation with diameter variations of $\pm 0.033\text{m}$ and width variations of $\pm 0.015\text{m}$. (e) Global scaling from 80% to 120% of baseline dimensions.

C Dataset Generation, Simulation, and Validation

C.1 Geometric Morphing via Reference Vehicle Architecture and Parametric Design

We develop a parametric geometry generation framework using lattice deformation to create diverse automotive geometries from the open-source DriveAer vehicle model. Our approach employs FFD

techniques implemented through Python scripts on the Blender[®] platform, establishing a $32 \times 8 \times 8$ three-dimensional lattice structure that enables fine-grained control over local body regions while maintaining geometric integrity.

The implementation begins with bounding box analysis to determine spatial distribution of geometric features, establishing mapping relationships between local and global coordinate systems. The parametric framework divides the vehicle into nine key regions including front bumper, trunk, and roof, with regional deformation control achieved through predefined index grouping. We define 20 key parameters covering body morphology, dimensional scaling, and wheel configurations (Table A1). Parameter ranges reference production vehicle design standards and incorporate CFD simulation experience to set boundary conditions that ensure engineering feasibility, with diffuser inclination constrained within $\pm 10^\circ$ to balance downforce and drag requirements.

LHS generates 1000 parameter samples across engineering-reasonable ranges for statistical representativeness, producing thousands of unique geometric configurations. Each parameter corresponds to specific physical modifications: dimensional stretches (front bumper length $\pm 0.1\text{m}$), positional adjustments (trunk inclination $\pm 0.05\text{m}$), size scaling (vehicle width $\pm 0.1\text{m}$), and angular rotations (windscreen inclination $\pm 0.05\text{m}$). Parameter distributions are detailed in Figure A1, with each configuration generating unique variants through nine independent deformation functions.

For cooling system integration—absent in previous datasets—we bind front grille, radiator, engine bay, and gearbox components to unified lattice deformers using LHS methodology. This coordinated deformation prevents spatial interpenetration while maintaining positional coherence among complex internal components during geometric transformations.

Precise wheel alignment employs a dedicated four-wheel positioning algorithm with predefined feature points including 3D coordinate systems for all four wheels. The algorithm dynamically calculates axle positions by tracking spatial transformations of these feature points during deformation, ensuring geometric matching between axles and deformed body while maintaining lattice deformation controllability and avoiding component distortion issues common in traditional morphing techniques.

This lattice deformation approach reduces geometric generation time by over 90% compared to traditional CAD parametric methods while preserving industrial-grade surface quality. The systematic parameterization via FFD maintains geometric continuity, topological consistency, and engineering feasibility while establishing a deformable configuration space rooted in industrial design practices. As illustrated in Figures A2 to A4, the method preserves aerodynamic baseline characteristics of the reference vehicle while enabling parametric integration of engineering details such as cooling systems and chassis components, generating complex geometric interactions including length-width coupling variations that provide foundation for neural networks to capture nonlinear flow features.

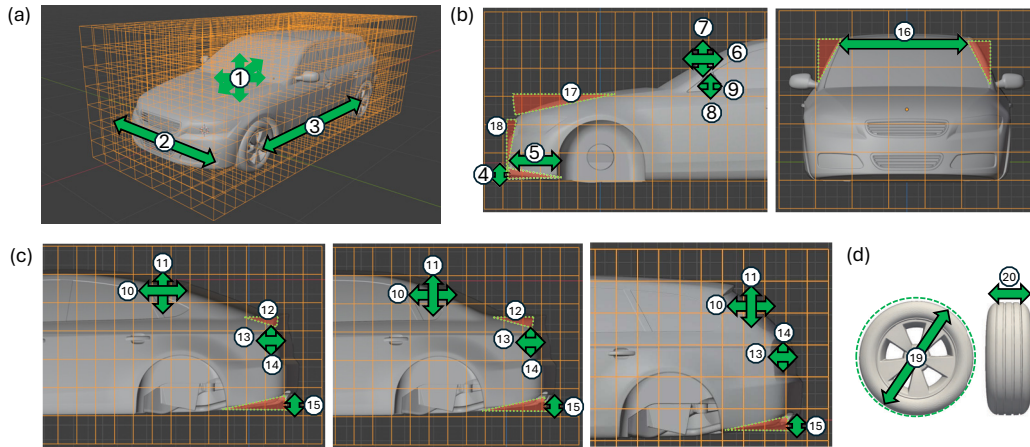


Figure A2: **Lattice deformation framework for DrivAer vehicle configurations.** (a) Deformation lattice structure with vehicle scaling, width, and length adjustment parameters. (b) Front vehicle section parameters controlling bumper and forward geometry modifications. (c) Rear section parameter sets for three vehicle types: fastback (left), notchback (center), and estateback (right) configurations. (d) Wheel geometry parameters for tire diameter and width variations.

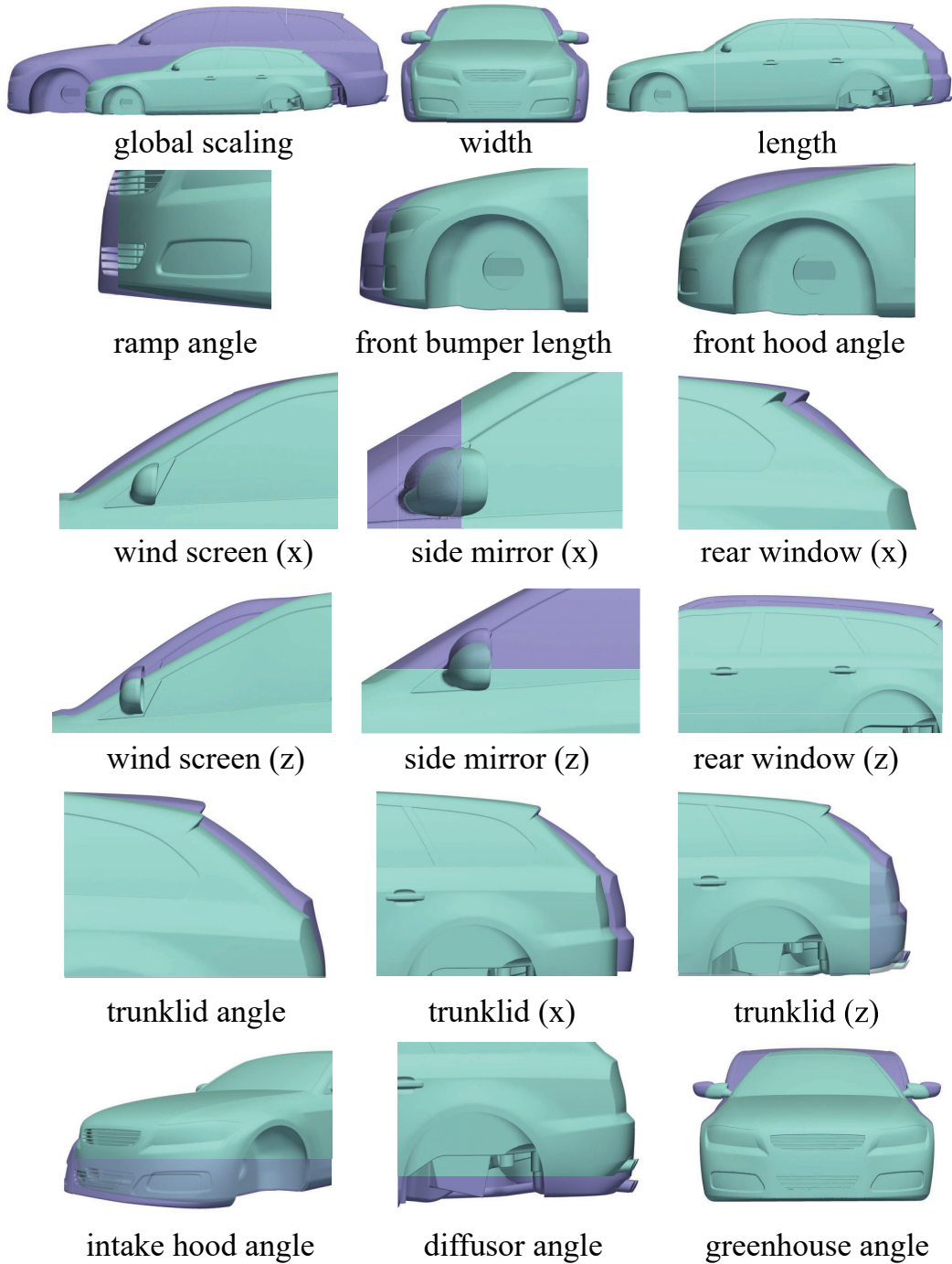


Figure A3: **Individual parameter deformation visualization.** Lattice deformation results for single-parameter variations across minimum to maximum ranges. Blue models represent minimum parameter values, purple models represent maximum values. Wheel size parameters are excluded to emphasize vehicle body geometry modifications.

Future research will integrate RANS simulation results to establish high-dimensional mapping models between morphological parameters and aerodynamic coefficients, supporting development of intelligent aerodynamic optimization frameworks.

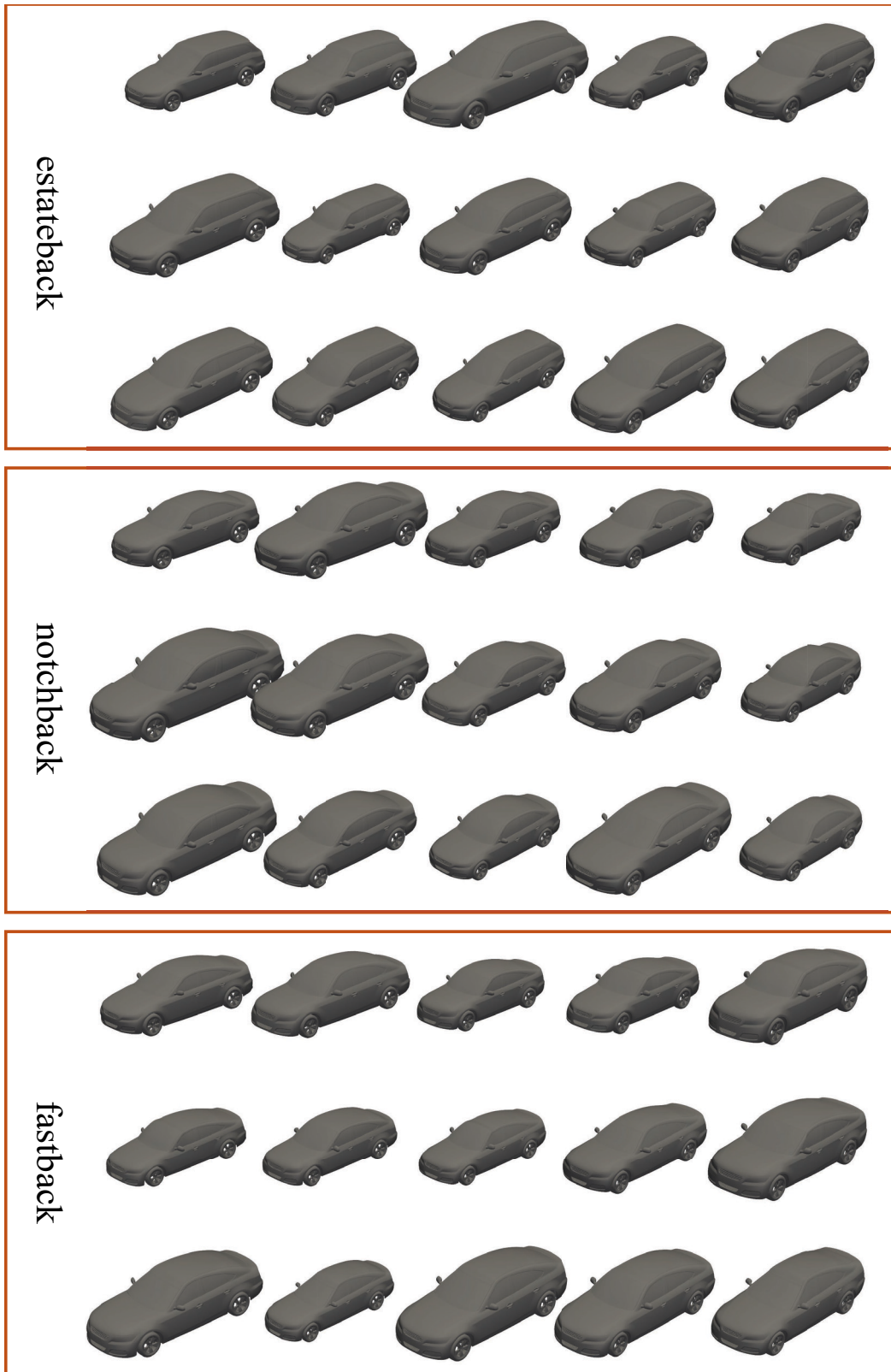


Figure A4: **Multi-parameter deformation examples by vehicle type.** Deformation variations for three rear configurations: estateback (top), notchback (middle), and fastback (bottom).

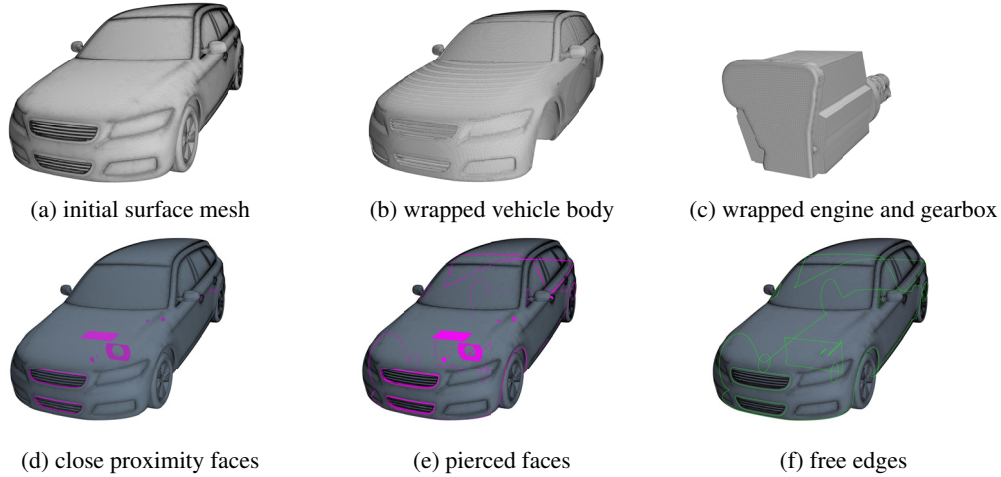


Figure A5: **Surface wrapping process using STAR-CCM+[®]**. Surface wrapping workflow applied to vehicle components: (a) Initial generated surface mesh with critical defects. (b) Wrapped vehicle body mesh. (c) Wrapped engine and gearbox components. The process addresses three fundamental surface errors: (d) close proximity faces, (e) pierced faces, and (f) free edges, with defective faces highlighted in purple and problematic edges in green.

C.2 Surface Wrapping Implementation Details

Generated vehicle geometries exhibit critical surface defects that prevent direct CFD computation, requiring mandatory surface wrapping preprocessing for all components. As illustrated in [Figure A5](#), we employ STAR-CCM+[®]'s surface wrapping functionality to independently process each component (body, drivetrain, wheels) of the initial surface mesh, addressing three fundamental mesh errors: (i) *Close Proximity* faces with minimum distances below 0.1 mm, (ii) *Pierced Faces* that create flow leakage paths, and (iii) *Free Edges* that compromise watertight geometric continuity.

Aerodynamically critical regions, including grilles and wheel arches, receive enhanced treatment through local refinement with 10% size reduction and curvature adaptation using a minimum spacing of 0.5 mm. Post-wrapping validation ensures aspect ratios remain below 20 and skewness angles exceed 10° across all mesh components. This optimized workflow achieves 98.6% defect reduction while maintaining 0.28 mm geometric fidelity, enabling reliable CFD simulations with aerodynamic feature resolution below 1 mm accuracy.

C.3 Regional Mesh

Mesh Generation Framework The parallel mesh generation employs a global base size $L_0 = 0.24$ m with surface growth rate 1.3, utilizing hexahedral elements enhanced by triangular surface reconstruction for geometric fidelity. Dual refinement strategies ensure critical feature capture: curvature refinement with 0.01 m deviation distance and 36 points per circle, plus proximity refinement for chassis, engine bay, and wheel components.

Volume growth adopts *very slow* mode with maximum element size constrained to 0.48 m ($200\%L_0$) and minimum surface size to 0.024 m ($10\%L_0$). Three-tiered volume controls implement progressive refinement: Block 2 at 0.06 m ($25\%L_0$), Block 3 matching L_0 , and Block 4 at 0.024 m ($10\%L_0$).

Surface and Boundary Layer Treatment Vehicle body surfaces receive enhanced treatment with target surface size 0.0144 m ($6\%L_0$) and minimum size 0.0096 m ($4\%L_0$). Prism layers follow geometric progression with first-layer height $h_1 = 0.005$ mm, expansion ratio $r = 1.2$, and 8-layer configuration yielding total thickness $H = h_1 \cdot (r^8 - 1)/(r - 1) \approx 0.114$ mm. External boundaries (inlet/outlet/side/top) utilize coarser 0.48 m elements without prism layer refinement.

Quality Control and Validation Mesh quality maintains triangular surface elements above a 0.05 minimum threshold with boundary layer resolution enforcing $y^+ > 30$ using blended wall functions.

Mesh independence verification employs three resolutions—coarse (0.48 m), medium (0.24 m), and fine (0.12 m)—validated through drag coefficient and flow field analysis.

C.4 Navier–Stokes Equations and Turbulence Model

Reynolds-Averaged Navier-Stokes Approach Automotive external aerodynamics simulation requires appropriate turbulent flow treatment. While Direct Numerical Simulation (DNS) captures all turbulence scales, it demands prohibitive computational resources for practical automotive applications. This study employs the RANS approach, modeling time-averaged flow fields at significantly reduced computational cost compared to DNS or Large Eddy Simulation (LES) alternatives.

The RANS equations derive from Reynolds decomposition, where each flow variable decomposes into mean and fluctuating components: $\phi = \bar{\phi} + \phi'$. For steady-state incompressible flow, the governing equations become:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \bar{\mathbf{v}}) = 0, \quad (\text{A1})$$

$$\frac{\partial}{\partial t}(\rho \bar{\mathbf{v}}) + \nabla \cdot (\rho \bar{\mathbf{v}} \otimes \bar{\mathbf{v}}) = -\nabla \bar{p} + \nabla \cdot (\bar{\mathbf{T}} + \mathbf{T}_{\text{RANS}}), \quad (\text{A2})$$

where ρ , $\bar{\mathbf{v}}$, \bar{p} , and $\bar{\mathbf{T}}$ represent air density, mean velocity vector, mean pressure, and mean viscous stress tensor, respectively. The Reynolds stress tensor \mathbf{T}_{RANS} requires turbulence modeling closure:

$$\mathbf{T}_{\text{RANS}} = -\rho \begin{pmatrix} \overline{u'u'} & \overline{u'v'} & \overline{u'w'} \\ \overline{u'v'} & \overline{v'v'} & \overline{v'w'} \\ \overline{u'w'} & \overline{v'w'} & \overline{w'w'} \end{pmatrix}, \quad (\text{A3})$$

where u' , v' , w' represent velocity fluctuations in the x -, y -, z -directions, respectively.

SST k - ω Turbulence Model This study employs the SST k - ω turbulence model (Menter, 1994), solving transport equations for turbulent kinetic energy k and specific dissipation rate ω . This model provides superior performance for boundary layers under adverse pressure gradients and applies throughout the boundary layer without wall-distance computation, yielding a reliable approximation for automotive aerodynamics.

The transport equations are formulated as:

$$\frac{\partial}{\partial t}(\rho k) + \nabla \cdot (\rho k \bar{\mathbf{v}}) = \nabla \cdot [(\mu + \sigma_k \mu_t) \nabla k] + P_k - \rho \beta^* f_{\beta^*} (\omega k - \omega_0 k_0), \quad (\text{A4})$$

$$\frac{\partial}{\partial t}(\rho \omega) + \nabla \cdot (\rho \omega \bar{\mathbf{v}}) = \nabla \cdot [(\mu + \sigma_\omega \mu_t) \nabla \omega] + P_\omega - \rho \beta f_\beta (\omega^2 - \omega_0^2), \quad (\text{A5})$$

$$\mu_t = \min \left(\frac{1}{\omega}, \frac{1}{F_2 S} \right), \quad (\text{A6})$$

$$\phi = F_1 \phi_1 + (1 - F_1) \phi_2, \quad (\phi = \sigma_k, \sigma_\omega, \beta), \quad (\text{A7})$$

where μ and μ_t denote dynamic and turbulent eddy viscosity, f_{β^*} and f_β represent free-shear and vortex-stretching modification factors, k_0 and ω_0 are ambient turbulence values (Spalart and Rumsey, 2007), and F_1 , F_2 are blending functions connecting inner and outer boundary layer regions.

Model parameters are: $\sigma_{k1} = 0.85$, $\sigma_{k2} = 1.0$, $\sigma_{\omega1} = 0.5$, $\sigma_{\omega2} = 0.856$, $\beta_1 = 0.075$, $\beta_2 = 0.0828$. Following the Boussinesq hypothesis, the Reynolds stress tensor is modeled as:

$$\mathbf{T}_{\text{RANS}} = \mu_t \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) - \frac{2}{3} \rho k \delta_{ij}, \quad (\text{A8})$$

providing complete system closure.

Porous Media Treatment Vehicle cooling components (radiators, condensers) are modeled as porous media due to their complex internal geometries that are too fine for individual meshing. The void fraction ε_g represents the geometry at coarse mesh scales, while momentum exchange is captured through body force terms.

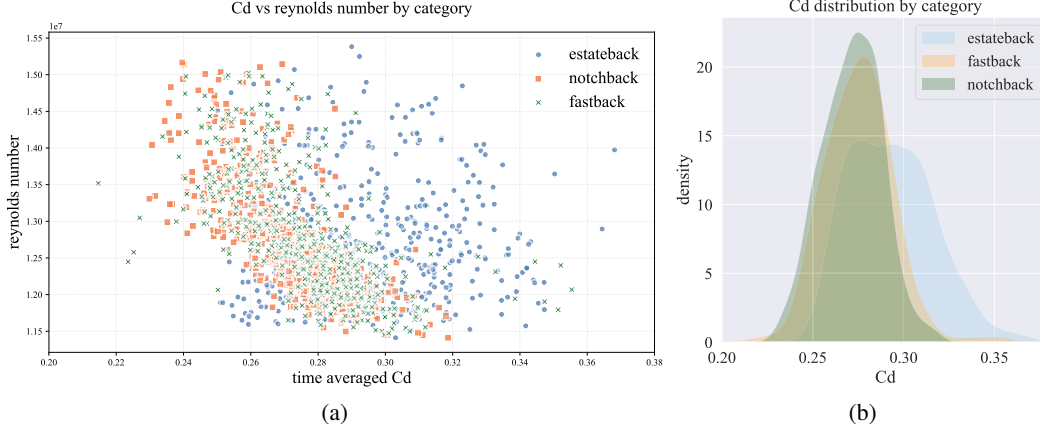


Figure A6: **Aerodynamic performance comparison of three rear configurations.** (a) Drag coefficient (C_D) versus Reynolds number (Re) with 95% confidence intervals (shaded regions). (b) Probability density distributions of C_D values highlighting modal characteristics. Estateback (blue), notchback (orange), and fastback (green) configurations are consistently color-coded across subfigures.

The governing equations for porous media become:

$$\frac{\partial(\varepsilon_g \rho)}{\partial t} + \nabla \cdot (\rho \bar{\mathbf{v}}) = 0, \quad (\text{A9})$$

$$\frac{\partial(\rho \bar{\mathbf{v}})}{\partial t} + \nabla \cdot (\rho \bar{\mathbf{v}} \otimes \bar{\mathbf{v}}) = -\nabla p + \nabla \cdot (\bar{\mathbf{T}} + \mathbf{T}_{\text{RANS}}) + \mathbf{F}_p, \quad (\text{A10})$$

where the flow resistance \mathbf{F}_p is given by:

$$\mathbf{F}_p = -(\mathbf{P}_v + \mathbf{P}_i |\bar{\mathbf{v}}|) \cdot \bar{\mathbf{v}}, \quad (\text{A11})$$

with \mathbf{P}_v and \mathbf{P}_i representing viscous and inertial resistance tensors. The superficial velocity $\bar{\mathbf{v}}$ relates to the actual pore velocity \mathbf{v}_g through $\bar{\mathbf{v}} = \varepsilon_g \mathbf{v}_g$.

Further implementation details are available in the [STAR-CCM+® Documentation](#).

C.5 Computational Infrastructure

Dataset generation utilized a high-performance computing cluster with 100 nodes, each equipped with Intel® Xeon® Gold 6148 processors, consuming approximately 1,080,000 core-hours. Computational efficiency was optimized through the AMG linear solver (Stüben, 2001), which accelerates convergence while reducing memory requirements.

C.6 Simulation Result

The aerodynamic behavior of three rear configurations (estateback, notchback, fastback) is quantified in [Figure A6](#), with comprehensive flow field visualizations including pressure distributions ([Figure A8](#)), wall shear stress patterns ([Figure A9](#)), and velocity profiles ([Figure A10](#)).

Reynolds Number Dependencies Drag characteristics exhibit distinct Reynolds number trends across configurations. The notchback configuration demonstrates consistent C_D reduction with increasing Re throughout the observed range, while the fastback shows progressive C_D decrease across its operational regime. In contrast, the estateback maintains relatively uniform C_D values without significant monotonic variation, though with broader dispersion compared to other configurations.

Statistical Characteristics The estateback exhibits bimodal density peaks at $C_D = 0.31$ and 0.35 , contrasting with unimodal distributions of notchback ($\mu = 0.28$, IQR = 0.04) and fastback ($\mu = 0.33$). The notchback's narrow interquartile range confirms superior aerodynamic consistency, while the fastback's right-skewed distribution reflects intermittent high-drag flow states.

C.7 Computational Validation and Quality Assurance

Automated Quality Control A comprehensive quality assessment framework with specific acceptance criteria ensures dataset integrity. Automated filtering verifies that physical quantities remain within engineering ranges: pressure values (volumetric and surface) and wall shear stress do not exceed $\pm 2 \times 10^4$ Pa. Simulations exhibiting numerical anomalies are automatically excluded from the dataset.

Convergence Criteria All cases undergo minimum 1,000 iteration steps with convergence determined by C_D stabilization within $\pm 1\%$ over the final 100 iterations. A hybrid convergence metric considers both relative C_D change and asymptotic behavior of solution residuals across continuity, turbulence, and momentum equations. Cases with unresolved oscillatory behavior or non-physical trends are identified through spectral analysis and excluded.

Mesh and Solution Quality Mesh quality verification ensures appropriate wall treatment application, with all simulations maintaining wall y^+ values outside the buffer layer for accurate near-wall flow resolution using blended wall functions. Meshes with significant skewness or aspect ratio defects undergo refinement or exclusion. Final quality control discards simulations with the lowest 5% composite quality scores, retaining only the top 60% of simulations based on computational quality metrics for subsequent analysis.

C.8 Differences Between Simulation and Experiments

Geometric Configuration Differences The primary geometric differences between simulation and experimental setups concentrate on wheel mounting systems and radiator representation. The front wheels utilize MacPherson strut-type mounting assemblies, while the rear wheels employ solid axle configurations. The experimental radiator geometry is unavailable in the CAD model and is instead represented through porous media modeling with pressure drop characteristics derived from hexagonal aluminum honeycomb and perforated sheet specifications.

The experimental model features five-spoke wheels with detailed tire tread patterns, symmetric mirror assemblies, comprehensive chassis geometry, and three rear configurations: fastback, notchback, and estateback. A precisely machined floor section maintains 4 mm wheel clearances, while detailed engine bay components include integrated cooling systems and spotlights. Wheel assembly tolerances are maintained within ± 0.5 mm.

Physical Modeling Validation Table 2 presents drag coefficient comparison between experimental measurements and `DrivAerStar` simulation results. All three rear configuration coefficients have been calibrated to approximate open-road operating conditions. The primary error sources include: flow regime complexities around the four-wheel assembly, pressure drop modeling through radiator porosity representation, and inherent turbulence model limitations in capturing separation and reattachment phenomena.

The validation demonstrates acceptable agreement between computational and experimental results, with discrepancies primarily attributed to geometric simplifications in the cooling system representation and the challenge of accurately modeling complex wheel-ground interactions in the computational domain.

D Benchmark Setup and Evaluation

D.1 Evaluation Metrics

Relative L_2 Loss (ϵ) The relative L_2 error serves as the primary evaluation metric for flow field regression across all experimental configurations. To ensure comparability between different physical variables, each flow field variable ϕ (velocity, pressure, wall shear stress) undergoes standardization using dataset-wide statistics:

$$\hat{\phi} = \frac{\phi - \mu_\phi}{\sigma_\phi}, \quad (\text{A12})$$

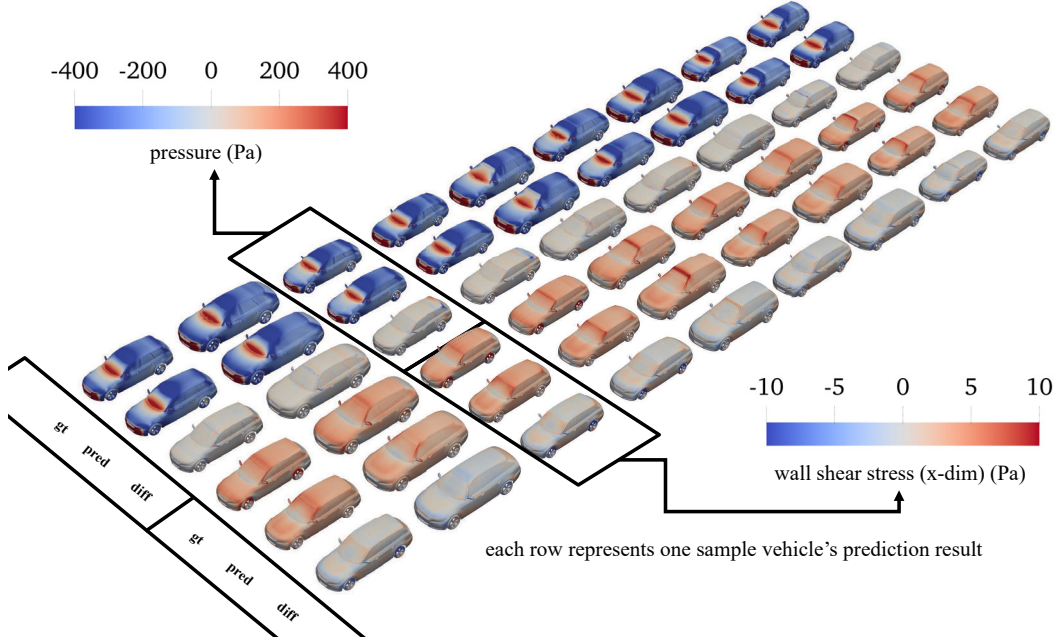


Figure A7: **Benchmark results visualization.** Nine testing cases from the single-vehicle setup demonstrate model performance across unique geometric configurations. Each row presents pressure fields (left) and wall shear stress results (right), with ground truth CFD data, machine learning predictions, and difference maps arranged from left to right within each prediction task.

where μ_ϕ and σ_ϕ represent the mean and standard deviation computed over the entire dataset. The normalized relative L_2 error is then defined as:

$$\epsilon = \frac{\|\hat{\phi}_{\text{pred}} - \hat{\phi}_{\text{true}}\|_2}{\|\hat{\phi}_{\text{true}}\|_2}, \quad (\text{A13})$$

where $\hat{\phi}_{\text{pred}}$ and $\hat{\phi}_{\text{true}}$ denote standardized predicted and ground-truth flow fields, respectively. This normalization eliminates scale discrepancies and aligns with established practices in industrial CFD model evaluation.

Drag Coefficient Counts Error Automotive industry practice quantifies drag coefficient accuracy using “counts error,” where 1 count equals 0.001 in C_D units. For example, if the simulated C_D is 0.300 and the predicted value is 0.320, the count error is 20 counts. This metric provides intuitive error quantification directly relevant to vehicle development targets.

D.2 Benchmark Result

Table A2 presents a systematic performance comparison between Transolver, GNOT, and PointNet frameworks across aerodynamic prediction tasks, revealing critical insights into their respective capabilities.

Single-Vehicle Specialization Transolver achieves superior performance in estateback-specific training, recording the lowest validation losses at 1,200 samples: pressure (0.1996), WSS (0.7953), and C_D (0.0266). The framework demonstrates strong scalability, with C_D prediction error decreasing 15.4% when expanding training data from 400 to 1,200 samples. GNOT shows substantial improvement in high-data regimes, with 1,200-sample C_D validation loss (0.0357) approaching Transolver’s performance, indicating enhanced learning capacity with sufficient data. PointNet exhibits inconsistent convergence behavior, showing pressure validation loss degradation (3.1% increase from 0.2991 to 0.2984) despite tripling training data. Single-vehicle benchmark results are visualized in **Figure A7**.

Multi-Vehicle Generalization Transolver maintains robust performance across vehicle configurations, showing minimal WSS validation loss degradation (2.9%) between single-vehicle (0.7953)

Table A2: **Training and validation loss comparison between Transolver and GNOT.** Loss curves for pressure, wall shear stress (WSS), and drag coefficient (C_D) prediction tasks. Training configuration: 500 samples per vehicle type (estateback, fastback, notchback) for 1,500 total samples. All loss values are normalized by initial conditions for comparative analysis.

(a) Transolver							
Category	Configuration	Training Loss			Validation Loss		
		Pressure	WSS	C_D	Pressure	WSS	C_D
Single-Vehicle	400×Estateback	0.2112	0.7841	0.0305	0.2173	0.7889	0.0314
	800×Estateback	0.2050	0.7793	0.0296	0.2086	0.7837	0.0283
	1200×Estateback	0.1969	0.7881	0.0315	0.1996	0.7953	0.0266
Multi-Vehicle	133×3 Vehicles	0.2076	0.8050	0.0421	0.2194	0.8049	0.0375
	266×3 Vehicles	0.1967	0.8130	0.0335	0.2075	0.8170	0.0335
	400×3 Vehicles	0.1957	0.7973	0.0333	0.2074	0.7871	0.0286

(b) GNOT							
Category	Configuration	Training Loss			Validation Loss		
		Pressure	WSS	C_D	Pressure	WSS	C_D
Single-Vehicle	400×Estateback	0.3250	0.9619	0.1144	0.3250	0.9619	0.1144
	800×Estateback	0.2185	0.7991	0.0463	0.2254	0.7986	0.0516
	1200×Estateback	0.1960	0.8135	0.0032	0.2087	0.8217	0.0357
Multi-Vehicle	133×3 Vehicles	0.2069	0.7802	0.0301	0.2122	0.7795	0.0344
	266×3 Vehicles	0.1968	0.7909	0.0284	0.2002	0.7920	0.0309
	400×3 Vehicles	0.1954	0.8171	0.0282	0.1964	0.8124	0.0315

(c) PointNet							
Category	Configuration	Training Loss			Validation Loss		
		Pressure	WSS	C_D	Pressure	WSS	C_D
Single-Vehicle	400×Estateback	0.2805	0.6460	0.0627	0.2991	0.6547	0.0502
	800×Estateback	0.2549	0.6755	0.0376	0.2787	0.6769	0.0499
	1200×Estateback	0.2717	0.6157	0.0571	0.2984	0.6273	0.0436
Multi-Vehicle	133×3 Vehicles	0.2971	0.6716	0.0722	0.3201	0.6767	0.0536
	266×3 Vehicles	0.2731	0.6875	0.0585	0.3007	0.7041	0.0653
	400×3 Vehicles	0.2872	0.6316	0.0589	0.3030	0.6560	0.0757

and 400×3 multi-vehicle (0.7871) scenarios. GNOT demonstrates superior cross-vehicle generalization, with 400×3 multi-vehicle C_D validation loss (0.0315) outperforming single-vehicle baseline (0.0357) by 11.8%, suggesting inherent architectural advantages for heterogeneous geometries. PointNet struggles with multi-task learning, exhibiting 41.7% higher average WSS validation loss compared to Transolver in equivalent configurations.

Task-Specific Analysis Wall shear stress prediction presents the greatest challenge across all frameworks, with the highest relative errors observed (Transolver: 0.7953, GNOT: 0.8124, PointNet: 0.6560 at 400×3 validation). Drag coefficient predictions reveal framework-dependent characteristics: Transolver achieves ultra-low errors (0.0266) through specialized training, while GNOT’s physics-informed architecture enables superior error consistency ($\pm 6.2\%$ variation) across vehicle configurations.

Sample Efficiency Analysis Transolver reaches 90% peak performance with 800 samples, showing marginal improvement ($\approx 2\%$) at 1,200 samples. GNOT requires 1,200 samples for comparable accuracy, while PointNet exhibits negative scaling beyond 800 samples, with 1,200-sample training yielding 4.3% higher pressure loss than the 800-sample case, indicating architectural limitations in leveraging large datasets.

D.3 Scaling Analysis with Training Sample Size

We conducted comprehensive scaling experiments using Transolver in multi-vehicle configurations to assess training dataset size impact on drag coefficient prediction. The analysis utilized 12,000 total samples partitioned into training subsets of varying sizes, with 150 independently generated samples for testing. This revised data splitting strategy ensures robust evaluation, with minor discrepancies from initial results attributed to refined subset partitioning. Validation loss and relative improvement (calculated as $\text{Improvement} = \frac{\text{Loss}_{400} - \text{Loss}_N}{\text{Loss}_{400}} \times 100\%$ against the 400-sample baseline) are presented in [Table 3](#).

Results demonstrate consistent C_D validation loss reduction with increased training data: doubling samples from 400 to 800 reduces loss by 10.67%, scaling to 1,200 samples achieves 23.73% reduction, and the full 12,000-sample dataset delivers 29.07% total improvement. This trend confirms Transolver’s effective data scaling properties, where larger training subsets directly translate to enhanced C_D prediction accuracy. The substantial accuracy gains achieved with the 12,000-sample dataset provide meaningful improvements for practical CFD applications, where marginal C_D prediction error reductions drive significant advances in vehicle aerodynamic optimization workflows. Raw experimental data, including detailed loss curves and test sample distributions, are provided in supplementary materials to ensure reproducibility.

E Data Sources of Automotive CFD Datasets

The comparative metrics presented in [Table 1](#) were compiled from multiple sources. For the DrivAerNet++ dataset ([Elrefaie et al., 2024](#)), the wall y^+ range and the average C_D precision were derived from the published simulation cases, while the wind tunnel error was extracted from Table 8 of the original paper. For the DrivAerML dataset ([Ashton et al., 2024c](#)), the wall y^+ range was not reported in the available literature; the average C_D precision was estimated from Figure 13(a) of the original paper, and the wind tunnel error was obtained from Table 1 in its appendix.

F Experimental Setting and Details

Implementation Details To ensure experimental reproducibility, all random number generators were initialized with a fixed seed of 42. Models were trained and evaluated on the DrivAerStar dataset under two configurations: 1 Vehicle (single configuration) and 3 Vehicle (multi-configuration). To assess data efficiency, the training set size was varied across 400, 800, and 1,200 samples, while the validation and test sets were fixed at 150 samples each. All models were trained from scratch for up to 500 epochs to ensure convergence. Training was conducted using the Adam optimizer with an initial learning rate of 0.001, adjusted by a cosine annealing scheduler, and a batch size of 4 and 1 for training and validation, respectively.

Network Architecture: Transolver The Transolver architecture ([Wu et al., 2024](#)) consists of 4 encoding layers, each implementing multi-head attention with 4 heads. The model uses a hidden dimension of 64 and maintains an MLP ratio of 1 in its feed-forward networks. The network processes 7-dimensional spatial input data and generates 4-dimensional outputs. The architecture incorporates 5 downsampling operations, utilizes a reference size of 8, and processes 16 data slices in parallel. During training, gradient clipping with a maximum norm of 0.1 was applied to stabilize optimization, while a weight decay of 0.0001 was used for regularization.

Network Architecture: PointNet The PointNet architecture ([Qi et al., 2017](#)) processes irregular mesh data through a transformation network (T-Net) and hierarchical feature extraction. The input transformation network generates a 7×7 transformation matrix using three 1D convolutional layers with channel dimensions of 64, 128, and 256. The main network employs sequential 1D convolutional layers with batch normalization and ReLU activation, progressively increasing feature dimensions from 64 to 256 channels. A residual connection bridges the initial 64-channel features to the final representation, while point-wise fully connected layers map the 256-dimensional features to 4-dimensional outputs.

Network Architecture: GNOT The GNOT architecture (Hao et al., 2023) processes irregular mesh data using neural operator principles. The input transformation network processes 7-dimensional features through three 1D convolutional layers (64, 128, and 256 channels), followed by max pooling and fully connected layers to generate transformation matrices. The main network consists of sequential 1D convolutional layers with batch normalization and ReLU activation, progressively increasing feature dimensions from 64 to 128 and finally to 256 channels. A residual connection via 1×1 convolution bridges the initial 64-channel features to the final 256-channel representation. Point-wise fully connected layers conclude the architecture by mapping 256-dimensional features to the 4-dimensional output space.

G Limitations and Future Work

Geometric Representation While our dataset employs 20 deformation parameters, this parameterization remains insufficient to fully characterize complete vehicle geometries. The simplified geometric representation limits the model’s ability to capture fine-scale aerodynamic features that significantly impact flow behavior in real-world applications.

Simulation Fidelity Although our simulations incorporate many industrial practices, they cannot capture all real-world complexities, including dynamic effects, environmental conditions, and manufacturing tolerances. The steady-state RANS approach, while computationally efficient, may not fully resolve transient flow phenomena critical for certain aerodynamic assessments.

Model Performance Validation Our experiments confirm that data-driven scaling laws remain effective, with model performance improving consistently as training samples increase. However, the observed improvements may plateau beyond current dataset sizes, requiring investigation of alternative approaches for continued accuracy gains.

Future Directions Future work will focus on dataset enhancement through expanded geometric parameterization and foundation model development for industrial applications. We plan to explore physics-informed learning methods and data assimilation techniques incorporating wind tunnel experimental data to improve model performance in limited-data scenarios. Additionally, investigation of multi-fidelity approaches combining RANS, LES, and experimental data may enhance prediction accuracy while maintaining computational efficiency for practical automotive design workflows.

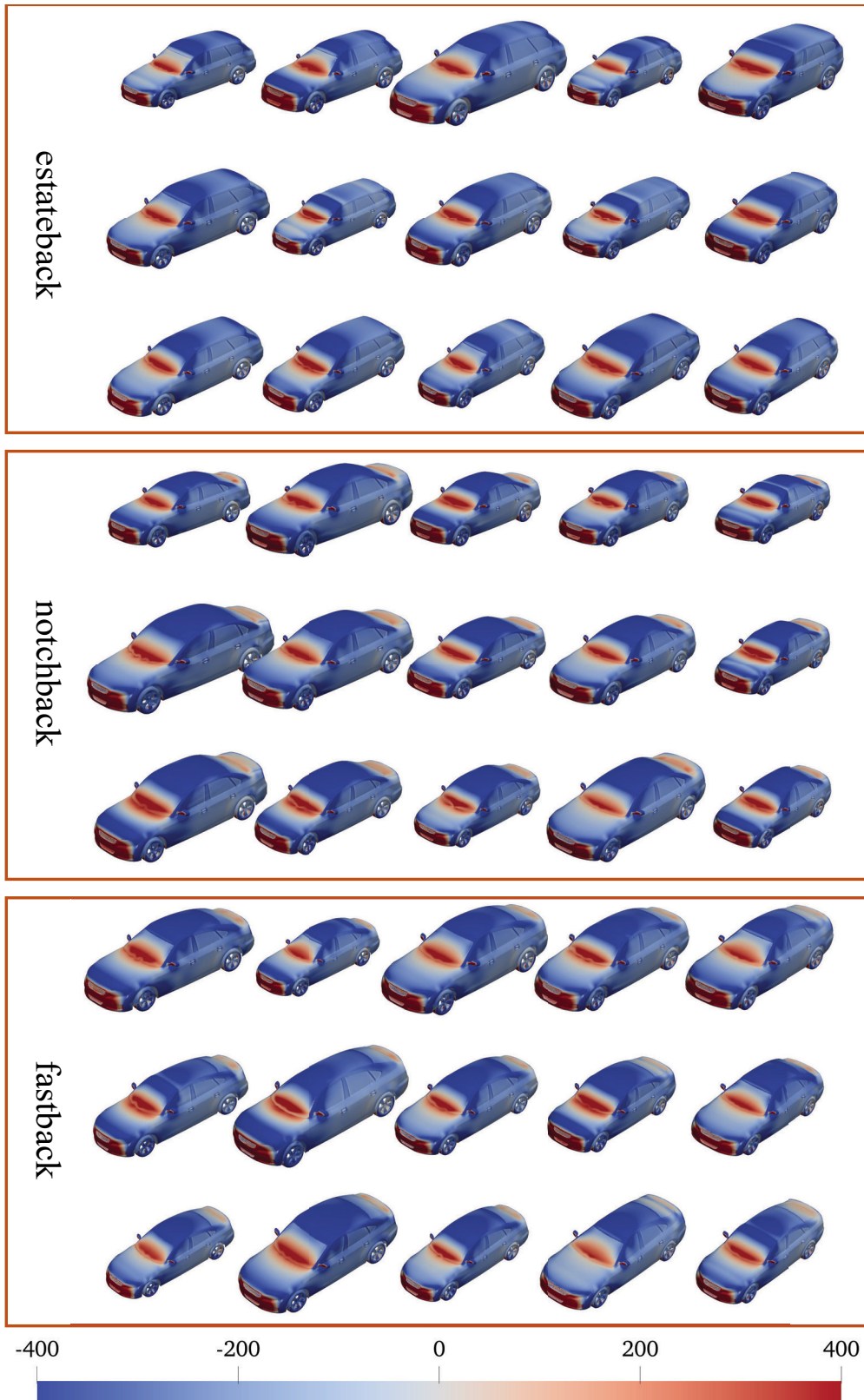


Figure A8: DrivAerStar simulation results: pressure field distributions (Pa).

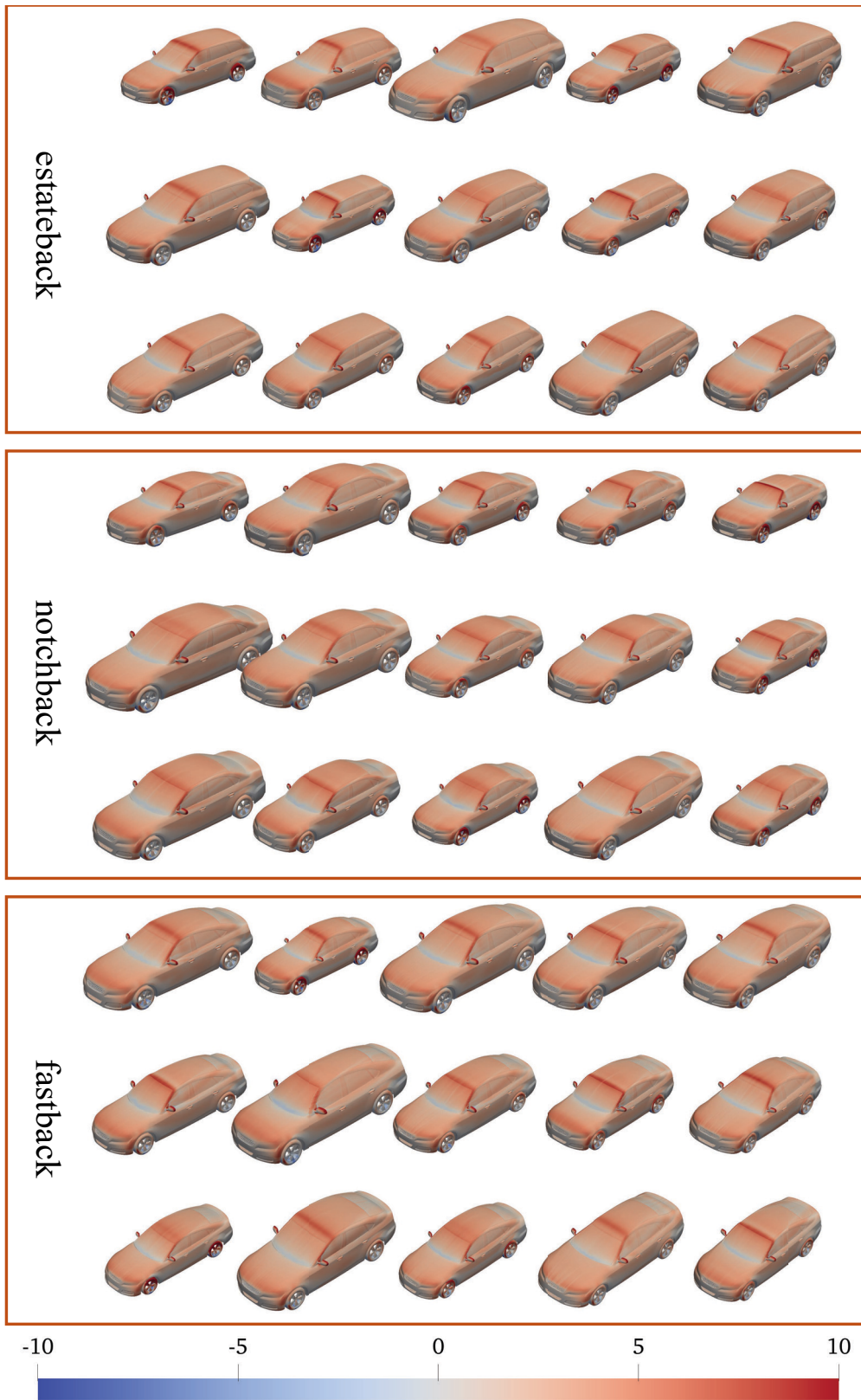


Figure A9: DrivAerStar simulation results: wall shear stress distributions (Pa).

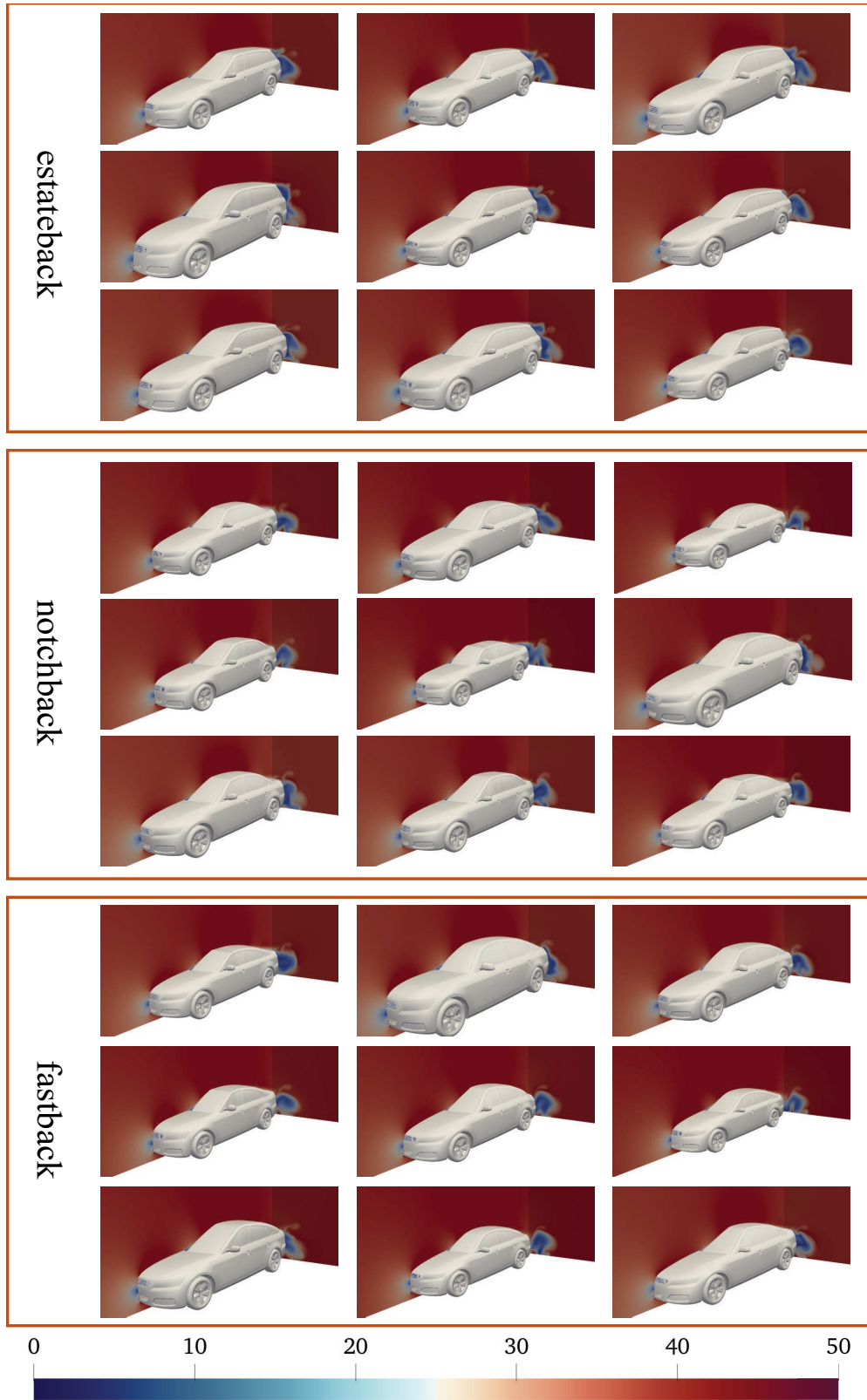


Figure A10: DrivAerStar simulation results: velocity field distributions (m/s)