DEAL: DENSITY-AGNOSTIC LEARNING FOR SAMPLING-STABLE GEOMETRY-TO-FIELD SURROGATES

Anonymous authors
Paper under double-blind review

ABSTRACT

Geometry-to-field surrogates predict physical fields from geometry and operating conditions, reducing the need for repeated numerical simulation. Because practical models receive finite meshes or point clouds rather than the underlying continuous domain, remeshing can redistribute local sampling density and connectivity without changing the physical case yet still alter predictions, a phenomenon we call *sampling-distribution sensitivity*. We introduce DeAL (Density-Agnostic Learning), a backbone-agnostic training principle that draws two fixed-budget views from a continuous inverse-density family for each case, supervises both predictions at common physical queries, and penalizes their disagreement. Dual supervision anchors both views to the physical target, while prediction consistency suppresses representation-dependent variation without changing the backbone or single-view inference. Across nine surrogate architectures and four physical systems, we find that DeAL reduces error and inconsistency among predictions by roughly half on average over a large number of matched comparisons when the same physical cases are evaluated under held-out spatial redistributions and independently generated remeshes, demonstrating transfer across distinct encoder designs and physical settings. Together, these results show that DeAL substantially improves predictions when the same physical case is presented through different numerical representations, without changing the underlying architecture or deployment path. In this way, our method increases robustness and reliability of neural surrogates as a key component for industrial applications.

1 INTRODUCTION

Mesh processing should be a numerical detail in predictive pipelines, not a source of physical disagreement. This is especially important for geometry-to-field surrogates, which accelerate prediction in complex domains (Wu et al., 2024a) while operating on finite point clouds or meshes produced by preprocessing the geometry. In common engineering workflows, a geometry may be regenerated through feature-aware decimation (Schroeder et al., 1992), quadric error metric (QEM) simplification (Garland & Heckbert, 1997), voxel-grid quadric clustering (Lindstrom, 2000), or other remeshing methods, each of which can redistribute samples and connectivity while closely preserving the surface. This input-redistribution can affect surrogate model's predictions. Even if training-aligned sampling is used for inputs, typical learning tasks are non-trivial and when the input sampling changes more substantially, very large deviations can occur: from our experience, inference errors can increase by more than $8\times$ for an otherwise unchanged objective. This is not merely a preprocessing artifact or a limitation of a particular architecture; it reflects a fundamental and largely unaddressed property of learning continuous physical operators from biased empirical representations, which can undermine the reliability of these models in practical settings.

We hypothesize that the source of this sensitivity lies in the mismatch between the continuous domain on which the physical problem is defined and the finite surface samples on which a surrogate operates. Remeshing can closely preserve the surface while changing where samples are concentrated. When an encoder aggregates features at these points, the resulting prediction can reflect the remesher's density pattern rather than only the physical geometry. Adding more points

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

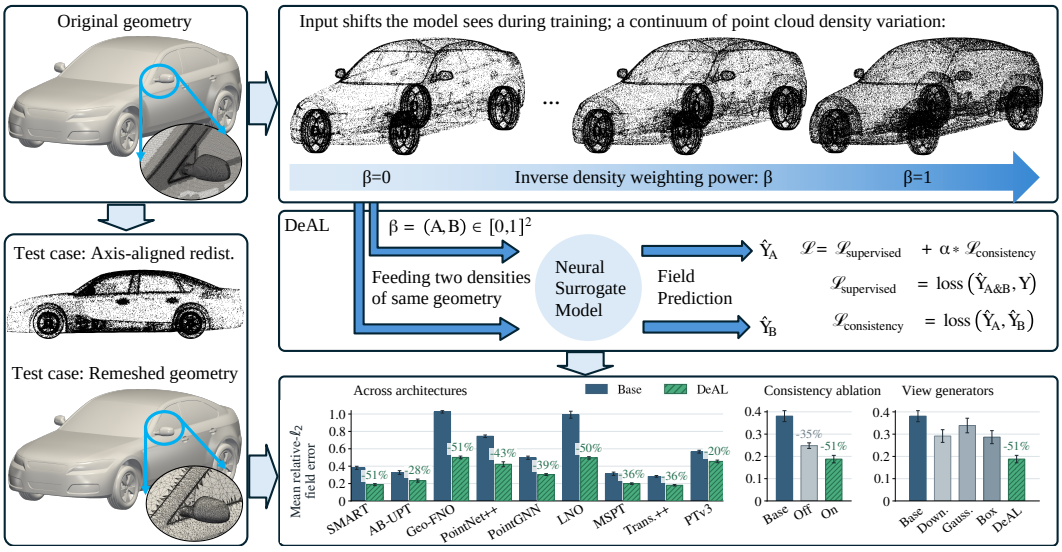


Figure 1: **DeAL overview.** Different point-sampled representations of the same physical geometry can yield inconsistent field predictions. DeAL trains an unchanged surrogate on paired density-varied views and enforces prediction consistency at shared physical queries. The car-aerodynamics results illustrate robustness gains under unseen spatial redistributions and remeshing, across architectures and relative to matched view generators.

need not remove this dependence if the relative sampling densities remain different. The problem can thus be stated generally: at fixed query locations Q and operating conditions λ, a surrogate F_θ may give unequal predictions for two samplings X_A, X_B drawn from the input-sampling distributions of views A and B over the same surface Γ: $F_\theta(X_A, Q, \lambda) \neq F_\theta(X_B, Q, \lambda)$.

This distinction also matters when choosing training perturbations: existing methods alter different properties of the input. Downsampling changes resolution (Qi et al., 2017), masked-point methods remove local support (Yu et al., 2022; Pang et al., 2022), and weighted local transformations deform neighborhoods (Kim et al., 2021). Consistency regularization encourages agreement under the perturbations treated as equivalent during training (Tarvainen & Valpola, 2017; Miyato et al., 2019). In this context, our work targets a fundamental challenge: *how can geometry-to-field surrogates be trained to retain accurate physical predictions across different sampling distributions of the same physical case?*

We address this challenge with DEAL (Density-Agnostic Learning), a backbone-agnostic training method built on two complementary components. For each case, we draw fixed-budget point sets from a continuous inverse-density family, evaluate both views at identical physical queries, supervise both predictions, and penalize their disagreement. Supervision anchors each view to the geometry-to-field target, while prediction consistency suppresses representation-induced variation. Our method requires neither a learned preprocessor nor a backbone modification, and inference remains unchanged. Figure 1 previews this approach and its empirical effect. Our central contributions are: **1.** We formalize sampling-distribution sensitivity as the dependence of a geometry-to-field surrogate on the empirical measure representing an otherwise fixed continuous surface. **2.** We introduce a novel continuous inverse-density view generator that varies empirical sampling measure at a fixed point budget while preserving the represented physical case. **3.** We pair this intervention with a backbone-agnostic learning objective that applies field supervision to both views and aligns their predictions at common physical queries, without changing the encoder, decoder, query mechanism, or single-view inference path. **4.** Across 36 matched evaluations of the conventional single-view baseline (Base) and DeAL spanning nine surrogate architectures and four physical systems, we show transfer to unseen redistributions and remeshes and isolate the roles of density range, estimator scale, prediction consistency, and view generation.

Section 2 formalizes the continuous-domain problem, Section 3 presents the DeAL training principle, and Section 4 positions it among relevant prior work. Sections 5 and 6 specify the

controlled evaluation protocol and examine transfer across architectures and physical systems, matched view generators, and the key contributions of density variation and prediction consistency. Section 7 discusses the findings, scope, and remaining directions.

2 Problem Formulation

2.1 Physical Operator and Representation Equivalence

For a fixed problem geometry and operating conditions, a physical field Y does not depend on how its boundary is sampled. Let $\Gamma \subset \mathbb{R}^3$ be the boundary surface of a physical case, λ its operating conditions, and $Q = (Q^s, Q^v)$ the surface and volume locations at which the fields are queried. The continuous target operator is

$$Y_\Gamma(Q, \lambda) = \mathcal{G}(\Gamma, \lambda)(Q). \tag{1}$$

To describe how Γ is sampled, let σ_Γ be its surface-area measure and $\bar{\sigma}_\Gamma = \sigma_\Gamma / \sigma_\Gamma(\Gamma)$ the corresponding uniform-area probability measure. A surrogate receives a finite input X containing surface points $\{x_i\}_{i=1}^n \subset \Gamma$ and mesh connectivity when provided. It predicts

$$\widehat{Y} = F_\theta(X, Q, \lambda). \tag{2}$$

The extra argument X describes how the surface is supplied to the model, not a change in the physical target. We call two inputs *representation-equivalent* when they describe the same physical case, including its boundary labels and conditions, but differ only in surface sampling or mesh connectivity. At fixed query locations Q, both have the same target in the form of Eq. (1). Let $\mathcal{R}(\Gamma)$ be the family of these representations. Our aim is $F_\theta(X, Q, \lambda) \approx \mathcal{G}(\Gamma, \lambda)(Q)$ for every $X \in \mathcal{R}(\Gamma)$. In particular, two such inputs $X_A, X_B \in \mathcal{R}(\Gamma)$, labeled A and B, should give consistent predictions:

$$F_\theta(X_A, Q, \lambda) \simeq F_\theta(X_B, Q, \lambda), \tag{3}$$

This is distinct from generalization across different shapes: a surrogate may predict new geometries well, yet remain sensitive to how each one is discretized (Sakarvadia et al., 2026).

2.2 How the Empirical Measure Enters a Surrogate

Representation-equivalent inputs can have different point distributions even though their physical target is fixed. A finite input $X_n = \{x_i\}_{i=1}^n$ defines an empirical probability measure μ_{X_n}, with δ_{x_i} denoting the Dirac measure at x_i. To model the effect of nonuniform sampling, suppose the points are independently sampled with spatial sampling density ρ relative to $\bar{\sigma}_\Gamma$, normalized by $\int_\Gamma \rho \, d\bar{\sigma}_\Gamma = 1$. The empirical measure and its weak limit are

$$\mu_{X_n} = \frac{1}{n} \sum_{i=1}^n \delta_{x_i}, \qquad \mu_{X_n} \xrightarrow[n\to\infty]{} \rho \, \bar{\sigma}_\Gamma. \tag{4}$$

Each point has weight $1/n$, so a more densely sampled region has greater representation in μ_{X_n} even when its physical area is unchanged. Uniform sampling by surface area has $\rho = 1$. This differs from uniform selection among source points (Section 3.1), which need not be area-uniform. Increasing n reduces random sampling noise, but if ρ remains nonuniform, the empirical measure converges to $\rho \, \bar{\sigma}_\Gamma$, not to uniform area. The input can therefore entangle geometry with where preprocessing placed samples.

This uneven sampling can affect a surrogate when it aggregates features from the sampled points. For example, at a query q, suppose an encoder forms a weighted average of learned point values $v_\theta(x_i)$ with positive affinities $a_\theta(q, x_i)$. Under the sampling law above, its limiting aggregate is

$$A_\rho(q) = \frac{\int_\Gamma a_\theta(q, x) v_\theta(x) \rho(x) \, d\bar{\sigma}_\Gamma(x)}{\int_\Gamma a_\theta(q, x) \rho(x) \, d\bar{\sigma}_\Gamma(x)}. \tag{5}$$

Normalizing the weights does not generally cancel ρ, so regions with more samples can retain greater influence even when the surface is unchanged. In different architectures, this aggregate effect of sampling density enters through different mechanisms such as neighborhood occupancy, pooling, hierarchical subsampling, voxelization, tokenization, or attention keys. While we do not claim that every architecture fails through the same internal mechanism, we target the observable and undesirable shared consequence: for some $X_A, X_B \in \mathcal{R}(\Gamma)$, $F_\theta(X_A, Q, \lambda) \neq F_\theta(X_B, Q, \lambda)$.

3 DENSITY-AGNOSTIC LEARNING

Below, we explain our approach to vary the sampling measure across paired views while fixing geometry, targets, and queries. Additionally, dual supervision preserves field accuracy, while prediction consistency suppresses representation-dependent residuals without modifying the backbone or inference path.

3.1 CONTINUOUS INVERSE-DENSITY VIEWS

Our approach starts with the available source surface cloud. A task-fixed normalization rule maps its coordinates to a dimensionless reference box. For each case, the resulting map T is held fixed across views. For each point x_i and its k nearest neighbors $\mathcal{N}_k(i)$, with $\bar{x}_i = T(x_i)$ a local Gaussian-kernel concentration score can be derived from kernel density estimation (KDE) (Silverman, 1986):

$$\widehat{\rho}_i = \frac{1}{k} \sum_{j \in \mathcal{N}_k(i)} \exp\left(-\frac{\|\bar{x}_i - \bar{x}_j\|_2^2}{h^2} \right), \qquad h^2 = \frac{1}{nk} \sum_{i=1}^{n} \sum_{j \in \mathcal{N}_k(i)} \|\bar{x}_i - \bar{x}_j\|_2^2. \tag{6}$$

Here h is a shared bandwidth and the neighborhood excludes the center point. $\widehat{\rho}_i$ is a monotone local concentration score rather than a surface-normalized density estimate; its normalization is not required as we only use relative weights below. This coordinate-only score constructs the views. It is never supplied to the surrogate as a material, boundary, or flow feature. KDE provides a smooth concentration estimate without requiring mesh connectivity or knowledge of the remesher, and therefore defines the same intervention for heterogeneous surface representations. We use $k = 16$ by default. Appendix Figure 8 and Table 11 test this neighborhood choice.

A view is a fixed-budget point-set representation of the source surface and is generated by sequential weighted sampling without replacement. Writing $\ell_i = \log \widehat{\rho}_i$, we stabilize the inverse-density weights by subtracting their maximum log-weight:

$$w_i^{(\beta)} = \max\left\{ \exp\left[-\beta \ell_i - \max_j(-\beta \ell_j) \right], 10^{-12} \right\}. \tag{7}$$

At each draw, let $\mathcal{U}_t$ be the set of points not selected previously. The next point $i \in \mathcal{U}_t$ is drawn with conditional probability $p_{i|\mathcal{U}_t}^{(\beta)} = w_i^{(\beta)} / \sum_{j \in \mathcal{U}_t} w_j^{(\beta)}$. Thus these normalized weights define a sequential sampling law; they are not marginal inclusion probabilities. At $\beta = 0$, Eq. (7) is uniform over the source cloud; increasing β progressively favors locally sparse regions. Thus β continuously controls intervention strength. Two independent draws X_A and X_B have the same budget and refer to the same Γ, but induce distinct empirical measures. The reference protocol samples $\beta_A, \beta_B \sim \mathcal{U}[0, 1]$.

3.2 PAIRED FIELD LEARNING ACROSS VIEWS

Both views pass through the same surrogate and are evaluated at identical queries. Let $Y = Y_\Gamma(Q, \lambda)$ be the physical target field, and write $\widehat{Y}_A = F_\theta(X_A, Q, \lambda)$ and $\widehat{Y}_B = F_\theta(X_B, Q, \lambda)$. Each branch receives Y, giving supervised losses

$$\mathcal{L}_{\text{sup}}^r = \ell(\widehat{Y}_r, Y), \qquad r \in \{A, B\}, \tag{8}$$

where ℓ aggregates the normalized surface and volume field discrepancies of the task. DeAL additionally matches predictions using a symmetric stop-gradient Smooth-ℓ_1 discrepancy (Huber, 1964),

$$\mathcal{L}_{\text{con}} = \frac{1}{2} \left[D(\widehat{Y}_A, \text{sg}[\widehat{Y}_B]) + D(\widehat{Y}_B, \text{sg}[\widehat{Y}_A]) \right]. \tag{9}$$

Here D is the fieldwise Smooth-ℓ_1 discrepancy over the shared queries and $\text{sg}[\cdot]$ stops gradients through its argument. The complete objective is a weighted sum,

$$\mathcal{L}_{\text{DeAL}} = \alpha_A \mathcal{L}_{\text{sup}}^A + \alpha_B \mathcal{L}_{\text{sup}}^B + \alpha_C \mathcal{L}_{\text{con}}, \qquad \alpha_A, \alpha_B, \alpha_C \geq 0. \tag{10}$$

The role of consistency is clearest under a squared discrepancy. Let $Z_B = F_\theta(X_B, Q, \lambda)$ be the prediction induced by a random view. For independent views B and B', the identity

$$\frac{1}{2} \mathbb{E}_{B,B'} \|Z_B - Z_{B'}\|_2^2 = \mathbb{E}_B \|Z_B - \mathbb{E}_{B'} Z_{B'}\|_2^2 \tag{11}$$

shows that half the expected pairwise squared disagreement equals prediction variance over the view distribution. Smooth-ℓ_1 supplies a robust local analogue. With positive supervised weights, the two supervised terms penalize a degenerate invariant solution: consistency is useful only while both outputs remain anchored to the physical field. In residual form, $\widehat{Y}_A - \widehat{Y}_B = (\widehat{Y}_A - Y) - (\widehat{Y}_B - Y)$, so DeAL suppresses residual components that change with representation.

Our method wraps an existing surrogate without altering its encoder, decoder, or query mechanism. The constraint acts only on outputs at common physical queries, so training with it requires no change to the backbone's internal representation or aggregation rule. The continuous view family and paired objective play complementary roles: the objective states what should remain stable, namely the predicted field at common physical queries. The view generator states which representation changes the model must learn to ignore. In our controlled comparisons, we use downsampling and local masking, retaining the same two-branch supervision-and-consistency structure as DeAL while changing only the view construction. Algorithm 1 consolidates the resulting training and inference procedure.

Algorithm 1 DeAL training and inference.

Prepare once. Normalize the surface coordinates, estimate each point's local concentration $\widehat{\rho}_i$, and cache it.
for each training step **do**
  Sample. Draw β_A, β_B and sample two independent m-point inverse-density views.
  Predict. Apply the shared surrogate to both views at the same physical queries Q.
  Learn. Supervise both predictions with Y, penalize their disagreement, and update θ using Eq. (10).
end for
Deploy normally. At inference, provide one available point cloud and predict once. No paired view or density calculation is required.

4 Related Work

The formulation above connects representation-equivalent physical inputs, inverse-density views, and consistency at common query locations. We position these choices relative to geometry-to-field learning, discretization-aware architectures, and consistency-based training.

Geometry-to-field learning. Neural operators learn maps between function spaces rather than solutions on a single fixed grid (Li et al., 2021; Lu et al., 2021; Kovachki et al., 2023). Extending this idea to irregular domains has motivated mesh graph networks (Brandstetter et al., 2022), latent and continuous-query operators (Wang & Wang, 2024; Wang et al., 2025), geometry-conditioned surrogates (Pepe et al., 2025), and scalable 3D surrogates (Holzschuh et al., 2026). These models differ in how they encode locality, aggregate geometry, and decode fields. We instead ask whether a fixed physical case receives a stable prediction across representation-equivalent finite point sets.

Discretization-aware geometric learning. Discretization mismatch can persist even when model parameters are nominally resolution independent. Related approaches include intrinsic surface operators (Sharp et al., 2022), density-compensated convolutions (Wu et al., 2019), and adaptive parametric representations (Kang et al., 2025). Set encoders provide permutation invariance, while point hierarchies explicitly address nonuniform local sampling. These architectural remedies are valuable, but they specialize the representation or aggregation rule. DeAL instead constrains the final field prediction across alternative empirical measures, making the same objective applicable to substantially different backbones.

Robustness and consistency. Consistency regularization encourages stable predictions under specified input or model perturbations. Its meaning, however, is determined by the chosen equivalence relation. Local patch augmentation modifies point neighborhoods (Sheshappanavar et al., 2021). Weighted local transformations provide another local intervention (Kim et al., 2021), while masked-point objectives remove local support (Yu et al., 2022; Pang et al., 2022). Random point subsets have also been used to certify robustness to point modification, insertion, and deletion (Liu et al., 2021). For a controlled comparison, we place downsampling, Gaussian-ball masking, and box masking inside the same paired field-supervision and consistency structure used by DeAL.

The defining difference is how the views are generated. This design distinguishes two contributions: paired learning defines stability at the predicted physical field, while continuous inverse-density sampling systematically varies the empirical measure without explicitly masking a surface region. To our knowledge, no prior geometry-to-field surrogate combines continuously-sampled inverse-density views, common-query physical-field supervision, and prediction consistency while leaving the backbone and single-view inference path unchanged. The experimental protocol, therefore, distinguishes the targeted empirical-measure shift from changes in geometry, global fidelity, and missing geometry components.

5 Experimental Protocol

We test whether DeAL transfers across architectures and physical systems, generalizes to unseen representation shifts, and outperforms matched view generators that alter resolution or local support. *Base* denotes conventional single-view training on the matched backbone.

5.1 Tasks and Architectures

We evaluate four geometry-to-field systems spanning external aerodynamics, internal hydrodynamics, nonlinear heat exchange, and magnetostatics. This selection varies geometry family, governing physics, and output structure while preserving the same learning problem: predicting physical fields from an unstructured surface representation. The fluid systems cover external and internal computational fluid dynamics (CFD), while the thermal and magnetic systems test whether the same representation sensitivity extends beyond CFD. For car aerodynamics, we use DrivAerML, a collection of parametrically morphed road-car geometries with high-fidelity CFD fields (Ashton et al., 2024). The surrogate receives a surface representation of the vehicle and predicts surface pressure, wall quantities, and volume flow variables at physical query locations. For pump hydrodynamics, we use SHIFT-Pump (Luminary Cloud, 2026), which contains varied centrifugal impeller–volute geometries and provider-generated CFD fields. Our third task is a nonlinear heat exchanger: a deterministic finite-element (FEM) benchmark in which geometry controls conduction between fixed-temperature internal channels and an exterior convection–radiation boundary. Our fourth task is C-core magnetostatics, a fixed-excitation finite-element problem on varied C-shaped ferromagnetic cores, for which the surrogate predicts the magnetic flux-density vector in the solid. Appendix B specifies the datasets, equations, boundary conditions, targets, and point budgets.

Across these tasks, we evaluate SMART (Hagnberger & Niepert, 2026), AB-UPT (Alkin et al., 2025), Geo-FNO (Li et al., 2023), PointNet++ single-scale grouping (SSG) (Qi et al., 2017), PointGNN (Shi & Rajkumar, 2020), the Latent Neural Operator (LNO) (Wang & Wang, 2024), MSPT (Curvo et al., 2026), Transolver++ (Luo et al., 2025), and Point Transformer V3 (PTv3) (Wu et al., 2024b). This range spans latent cross-attention, anchored branches, geometry-aware spectral operators, hierarchical point processing, radius-graph message passing, multiscale attention, and operator-style models. For every Base–DeAL pair, we hold the backbone, physical training cases, target normalization, point budget, and query coordinates fixed. We train Base for 300 epochs and DeAL for 150, as each DeAL epoch supervises two views.

5.2 Evaluation-Only Representation Shifts

The inverse-density family in Eq. (7) defines the paired training views. We reserve two shift families absent from training for evaluation. The first, axis-structured redistribution, samples points with probability proportional to $\sin^2(\pi z_i) + \epsilon$, where z_i is a normalized x or y coordinate and $\epsilon > 0$ prevents zero sampling probability. Applied along either axis, this produces Sine-x and Sine-y views that alter spatial support while preserving the surface.

The second family uses independent feature-aware, QEM, and voxel-grid remeshing to preserve topology and sharp surface features, minimize accumulated geometric error, and aggregate vertices within uniform spatial cells, respectively. We evaluate 5x and 10x reductions where applicable and sample fixed-budget point-cloud inputs. Across both shift families, only the encoder representation changes, while the physical case, targets, query coordinates, and query budgets remain fixed. Remesh fidelity is evaluated and visualized separately in Appendix Figure 9 and Table 17.

5.3 Evaluation and Controlled Studies

Let Y^s and Y^v denote the physical surface and volume targets, and let $\widehat{Y}^s$ and $\widehat{Y}^v$ be predictions at the same fixed query locations. For each domain $d \in \{s, v\}$ with C_d output channels, we compute a relative-ℓ_2 error independently for every channel and report

$$E = \frac{1}{2} \sum_{d \in \{s,v\}} \frac{1}{C_d} \sum_{c=1}^{C_d} \frac{\|\widehat{Y}^d_{:,c} - Y^d_{:,c}\|_2}{\|Y^d_{:,c}\|_2}, \tag{12}$$

where the norm spans query locations for one output channel. We average E over cases and paired encoder views. We report the mean and case-level standard deviation in plots and tables. We use paired bootstrap resampling over cases (Efron, 1979) to form 95% intervals for the Base–DeAL reduction in Table 14. Let $\bar{E}_m$ denote the mean of E over the matched cases and encoder views for method m. Relative reduction is used only as a paired annotation,

$$\Delta = 100\,(\bar{E}_{\text{Base}} - \bar{E}_{\text{DeAL}})/\bar{E}_{\text{Base}}, \tag{13}$$

so the magnitude of each error remains visible. We compare methods with identical cases, query coordinates, point budgets, and view seeds. Architecture comparisons use task-specific held-out test subsets. Cross-system summaries average paired percentage reductions rather than pooling absolute errors from fields with different physical scales. The view-generator controls replace inverse-density sampling with uniform downsampling, Gaussian-ball masking, or box masking, while retaining the same two supervised branches, consistency term, and physical query locations. They therefore isolate view-generator choice, distinguishing density redistribution from global resolution loss and localized deletion.

6 Results

We organize the results around the central hypothesis that representation-equivalent inputs should produce stable physical predictions. We first test whether DeAL reduces both shifted-input error and representation-induced disagreement, then isolate whether the gains depend on inverse-density view construction, prediction consistency, and continuous density variation.

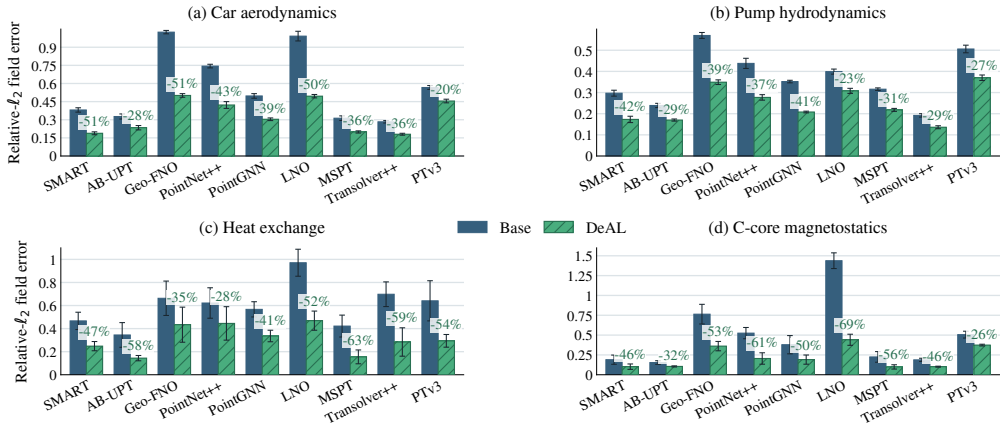


Figure 2: **DeAL reduces shifted-input error across architectures.** (a–d) Car aerodynamics, pump hydrodynamics, heat exchange, and C-core magnetostatics. For each architecture, bars report mean relative-ℓ_2 field error over the held-out shifts, and labels give the paired reduction from Base. DeAL results in error reductions of up to 69%.

6.1 DeAL Transfers Across Unseen Representation Shifts

Across all 36 task–architecture pairs in Figure 2, the Base shifted-input error exceeds its corresponding $\beta = 0$ reference, with a median shifted-to-reference ratio of 2.1×. DeAL

lowers shifted-input error in every pair, yielding a 42.4% reduction on average and individual reductions from 19.7% to 69.3%. For the SMART reference study, DeAL improves all 16 system–condition combinations spanning structured redistributions and remeshing, and every paired 95% reduction interval remains above zero (Appendix Table 14). The remeshing setup was validated using remesh-fidelity audits to ensure the inputs retain the underlying geometry (Appendix I). Figure 3(b–d) makes the aggregate result spatial: averaged across held-out shifts, DeAL contracts the dominant Base-error regions in every system. The strongest matched control reduces some of these errors, but its residual error remains visibly larger than DeAL's. Target-relative error alone, however, does not establish representation stability because equivalent inputs can have similar target errors while producing different predictions. Appendix E therefore measures direct prediction disagreement at identical physical queries. DeAL lowers disagreement in all 36 matched pairs by 49.8% on average. In every pair, disagreement decreases proportionally more than shifted-input error, showing that DeAL improves representation stability alongside accuracy, as sought in Eq. (3).

6.2 Effect of View Generator

Focusing on a single architecture (SMART), we compare DeAL against matched controls in Figure 3(e). The controls retain the same two-view supervision-and-consistency objective, changing only the view construction. DeAL achieves the lowest mean error in every system, reducing Base error by 46.4% on average across systems. Even selecting the strongest matched control separately within each system yields only a 23.5% reduction on average, compared with 46.4% for DeAL. Thus, the shared two-view objective does not account for the full gain. Within the tested controls, varying empirical sampling density at a fixed point budget is substantially more effective under the held-out representation shifts than reducing resolution or removing local support.

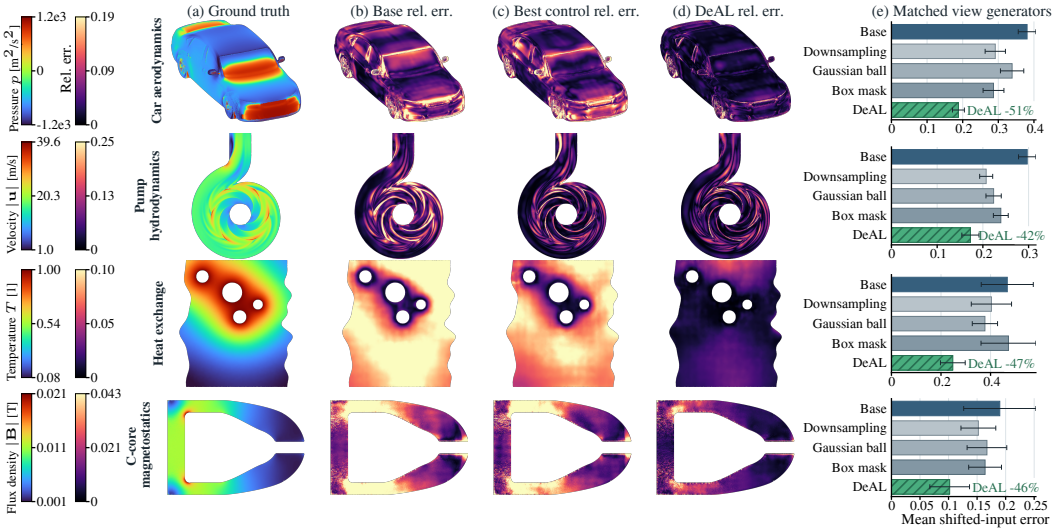


Figure 3: **Spatial errors and matched view controls across systems.** (a) Native fields. (b–d) Pointwise relative errors averaged over held-out shifts for Base, the lowest-error matched control selected per system (Appendix F), and DeAL. Across all four systems, DeAL suppresses a significant portion of Base-error region, while the strongest matched controls retain larger residual errors. (e) Other approaches for generating novel views of the input data yield substantially less improvement compared to the proposed method. Shown here for a fixed architecture (SMART), mean relative-ℓ_2 error on the held-out test subset; labels report DeAL's paired reduction.

6.3 Effect of Prediction Consistency and Continuous Density

To isolate the contribution of prediction consistency, Figure 4 retains two density-varied supervised views while switching only the consistency term. Density-varied supervision alone reduces Base error by 30.0–34.8% across systems, showing that a substantial part of the robustness gain is already obtained before prediction consistency is introduced. Relative to this otherwise identical

setting, adding prediction consistency lowers error by a further 16.9–24.4%, or 20.3% on average, and lowers error in all 16 held-out system–condition comparisons reported in Appendix Table 12. The full DeAL gain therefore cannot be attributed to density-varied dual supervision alone. The consistency term makes a distinct contribution after both views are already supervised. Appendix Figure 7 tests a separate question: whether the continuous density family matters beyond simply pairing two views. With supervision and consistency otherwise fixed, two $\beta = 0$ views and two $\beta = 1$ views reduce Base error by only 2.1% and 7.1% on average across systems, while the discrete endpoint pair $\beta \in \{0, 1\}$ reaches 27.2%. Independent $\beta \sim \mathcal{U}[0, 1]$ draws reach 46.4% and give the lowest error in every system. Thus, neither paired stochasticity at a fixed density nor exposure only to the two density extremes recovers the full benefit, supporting continuous coverage of intermediate empirical sampling measures.

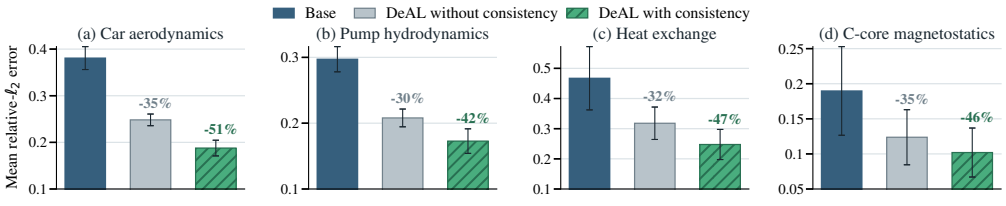


Figure 4: **Prediction-consistency ablation across systems, with SMART held fixed.** (a–d) Car aerodynamics, pump hydrodynamics, heat exchange, and C-core magnetostatics. Base uses one supervised view. Both DeAL variants use two density-varied supervised views, with prediction consistency disabled or enabled. Bars summarize held-out conditions. Enabling prediction consistency reduces residual error in every system, showing that it complements density-varied supervision.

7 Discussion and Limitations

The central implication of these experiments is that robustness to sampling changes is not obtained merely by exposing a surrogate to multiple views of the same geometry; it depends on which representation changes the model is asked to treat as equivalent. In our experiments, perturbations that reduce resolution or remove local support do not reproduce the behavior obtained by systematically varying the empirical sampling measure while preserving the physical case and point budget. This distinction suggests that the choice of training variation is itself part of defining the invariance we want the surrogate to learn. Within DeAL, the two components play complementary roles: supervision keeps each view anchored to the same physical field, while prediction consistency directly discourages representation-dependent variation between their outputs. Sampling density continuously, rather than at a single setting or only at its extremes, further exposes the model to a range of equivalent numerical representations instead of a small set of discrete cases. We therefore view DeAL as a practical way to encode representation equivalence during training, targeting a largely unaddressed failure mode of current geometry-to-field surrogate methodologies, while requiring no architectural or inference procedure changes.

Our evaluation deliberately isolates representation changes that preserve the physical case, geometry, and query correspondence, allowing us to attribute the observed effects to discretization rather than to a changed prediction problem. This framing also points to natural extensions: adaptive and time-varying meshes where sampling patterns may be coupled to refinement criteria or evolving states, and mesh-native surrogates where connectivity enters the computation directly. Another complementary direction is to study representation stability alongside geometric generalization. These directions build on the principle supported by our results: a surrogate's prediction should be governed by the underlying physical case, not by the particular discretization used to present it. By improving accuracy and prediction stability across such changes, DeAL brings geometry-to-field surrogates closer to reliable use in engineering workflows where designs are routinely re-meshed and resampled.

8 Ethics Statement

DeAL is intended for scientific surrogate modeling, where it can reduce the computational cost of evaluating geometry-dependent physical fields. Such surrogates should not replace validation by trusted numerical solvers, experiments, or established engineering safety processes in high-consequence decisions. In particular, sampling stability is not a certificate of physical accuracy outside the training distribution. We therefore state input-coverage limits and uncertainty-relevant diagnostics rather than presenting the method as a substitute for verification. The accompanying code and data documentation will follow dataset licenses, attribution requirements, and restrictions on releasing trained model weights.

9 Reproducibility Statement

Sections 2 and 3 specify representation equivalence, view construction, and objectives; Sections 5 and 6 specify evaluation conditions and reported metrics. Appendix A records numerical view construction, Appendix B gives task-specific budgets, Appendix C documents shared optimization and architecture configurations, and Appendix F reports the controlled ablations. Base and DeAL are trained independently from random initialization for 300 and 150 epochs, respectively. The canonical reference runs fix the random seed to 42; Appendix H reports the five-seed replication, including seed 42. We use a $k = 16$ coordinate-only KDE on each source surface and draw two independent weighted samples without replacement. The reported DeAL runs use only $\beta_A, \beta_B \sim \mathcal{U}[0, 1]$ during training; Sine-x, Sine-y, and independent remeshes are evaluation conditions.

Source code, generated datasets, and trained models will be published to ensure reproducibility.

10 Statement on the Use of AI Systems

Generative-AI tools assisted with software prototyping and review, experiment organization, and manuscript drafting and editing. They were not used as evidence, as a substitute for literature review, or to validate experimental claims. The authors manually reviewed the final code, equations, citations, figures, numerical results, and prose, and take responsibility for all content in the submission.

References

Benedikt Alkin, Maurits Bleeker, Richard Kurle, Tobias Kronlachner, Reinhard Sonnleitner, Matthias Dorfer, and Johannes Brandstetter. AB-UPT: Scaling neural CFD surrogates for high-fidelity automotive aerodynamics simulations via anchored-branched universal physics transformers. *Transactions on Machine Learning Research*, 2025. URL https://openreview.net/forum?id=nwQ8nitlTZ.

Neil Ashton, Charles Mockett, Marian Fuchs, Louis Fliessbach, Hendrik Hetmann, Thilo Knacke, Norbert Schonwald, Vangelis Skaperdas, Grigoris Fotiadis, Astrid Walle, Burkhard Hupertz, and Danielle Maddix. Drivaerml: High-fidelity computational fluid dynamics dataset for road-car external aerodynamics. *arXiv preprint arXiv:2408.11969*, 2024. URL https://arxiv.org/abs/2408.11969v2.

Johannes Brandstetter, Daniel Worrall, and Max Welling. Message passing neural pde solvers. In *International Conference on Learning Representations*, 2022.

Zhao Chen, Vijay Badrinarayanan, Chen-Yu Lee, and Andrew Rabinovich. Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pp. 794–803, 2018.

Pedro M. P. Curvo, Jan-Willem van de Meent, and Maksim Zhdanov. MSPT: Efficient large-scale physical modeling via parallelized multi-scale attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12924–12933, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Curvo_MSPT_Efficient_Large-Scale_Physical_Modeling_via_Parallelized_Multi-Scale_Attention_CVPR_2026_paper.html.

Bradley Efron. Bootstrap methods: Another look at the jackknife. *The Annals of Statistics*, 7(1):1–26, 1979. doi: 10.1214/aos/1176344552.

Michael Garland and Paul S. Heckbert. Surface simplification using quadric error metrics. In *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*, pp. 209–216, 1997.

Christophe Geuzaine and Jean-François Remacle. Gmsh: A three-dimensional finite element mesh generator with built-in pre- and post-processing facilities. *International Journal for Numerical Methods in Engineering*, 79(11):1309–1331, 2009. doi: 10.1002/nme.2579.

Tom Gustafsson and G. D. McBain. scikit-fem: A Python package for finite element assembly. *Journal of Open Source Software*, 5(52):2369, 2020. doi: 10.21105/joss.02369.

Jan Hagnberger and Mathias Niepert. SMART: Scalable mesh-free aerodynamic simulations from raw geometries using a transformer-based surrogate model. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*, 2026. URL https://arxiv.org/abs/2601.18707v2.

Benjamin Holzschuh, Georg Kohl, Florian Redinger, and Nils Thuerey. P3D: Highly scalable 3d neural surrogates for physics simulations with global context. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/bf67feee86c1711d63cb4e6dc9998702-Abstract-Conference.html.

Peter J. Huber. Robust estimation of a location parameter. *The Annals of Mathematical Statistics*, 35(1):73–101, 1964. doi: 10.1214/aoms/1177703732.

Namgyu Kang, Jaemin Oh, Youngjoon Hong, and Eunbyung Park. PIG: Physics-informed gaussians as adaptive parametric mesh representations. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/b5aa1fd43f59dae7fef9d199b09e26b5-Abstract-Conference.html.

Alex Kendall, Yarin Gal, and Roberto Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7482–7491, 2018.

Sihyeon Kim, Sanghyeok Lee, Dasol Hwang, Jaewon Lee, Seong Jae Hwang, and Hyunwoo J. Kim. Point cloud augmentation with weighted local transformations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 548–557, 2021.

Nikola Kovachki, Zongyi Li, Burigede Liu, Kamyar Azizzadenesheli, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Neural operator: Learning maps between function spaces with applications to PDEs. *Journal of Machine Learning Research*, 24(89):1–97, 2023. URL https://jmlr.org/papers/v24/21-1524.html.

Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. *International Conference on Learning Representations*, 2021.

Zongyi Li, Daniel Zhengyu Huang, Burigede Liu, and Anima Anandkumar. Fourier neural operator with learned deformations for PDEs on general geometries. *Journal of Machine Learning Research*, 24(388): 1–26, 2023. URL https://jmlr.org/papers/v24/23-0064.html.

Peter Lindstrom. Out-of-core simplification of large polygonal models. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, pp. 259–262, 2000.

Hongbin Liu, Jinyuan Jia, and Neil Zhenqiang Gong. PointGuard: Provably robust 3d point cloud classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6186–6195, 2021.

Qiang Liu, Mengyu Chu, and Nils Thuerey. ConFIG: Towards conflict-free training of physics-informed neural networks. In *International Conference on Learning Representations*, 2025.

Ilya Loshchilov and Frank Hutter. SGDR: Stochastic gradient descent with warm restarts. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=Skq89Scxx.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

Lu Lu, Pengzhan Jin, Guofei Pang, Zhongqiang Zhang, and George Em Karniadakis. Learning nonlinear operators via deeponet based on the universal approximation theorem of operators. *Nature Machine Intelligence*, 3(3):218–229, 2021.

Luminary Cloud. SHIFT-Pump: High-fidelity computational fluid dynamics dataset for centrifugal pump design and analysis. Hugging Face dataset, 2026. URL https://huggingface.co/datasets/luminary-shift/Pump/tree/8a332ffb079cbbc10a521112b8bcbbef0a03cf51. Revision 8a332ffb079cbbc10a521112b8bcbbef0a03cf51.

Huakun Luo, Haixu Wu, Hang Zhou, Lanxiang Xing, Yichen Di, Jianmin Wang, and Mingsheng Long. Transolver++: An accurate neural solver for PDEs on million-scale geometries. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 41432–41449, 2025. URL https://proceedings.mlr.press/v267/luo25o.html.

Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. Virtual adversarial training: A regularization method for supervised and semi-supervised learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(8):1979–1993, 2019.

Yatian Pang, Wenxiao Wang, Francis E. H. Tay, Wei Liu, Yonghong Tian, and Li Yuan. Masked autoencoders for point cloud self-supervised learning. In *European Conference on Computer Vision*, pp. 604–621, 2022.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Alberto Pepe, Mattia Montanari, and Joan Lasenby. Fengbo: A clifford neural operator pipeline for 3d PDEs in computational fluid dynamics. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/3711d4376db81f2d8e4177c6525a9b0a-Abstract-Conference.html.

Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Mansi Sakarvadia, Kareem Hegazy, Amin Totounferoush, Kyle Chard, Yaoqing Yang, Ian Foster, and Michael W. Mahoney. The false promise of zero-shot super-resolution in machine-learned operators. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=hkF7ZM7fEp.

Joachim Schöberl. NETGEN: An advancing front 2d/3d-mesh generator based on abstract rules. *Computing and Visualization in Science*, 1(1):41–52, 1997. doi: 10.1007/s007910050004.

William J. Schroeder, Jonathan A. Zarge, and William E. Lorensen. Decimation of triangle meshes. *ACM SIGGRAPH Computer Graphics*, 26(2):65–70, 1992. doi: 10.1145/142920.134010.

Nicholas Sharp, Souhaib Attaiki, Keenan Crane, and Maks Ovsjanikov. Diffusionnet: Discretization agnostic learning on surfaces. *ACM Transactions on Graphics*, 41(3), 2022. doi: 10.1145/3507905.

Shivanand Venkanna Sheshappanavar, Vinit Veerendraveer Singh, and Chandra Kambhamettu. PatchAugment: Local neighborhood augmentation in point cloud classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pp. 2118–2127, 2021.

Weijing Shi and Raj Rajkumar. Point-GNN: Graph neural network for 3d object detection in a point cloud. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1711–1719, 2020. doi: 10.1109/CVPR42600.2020.00178.

Bernard W. Silverman. *Density Estimation for Statistics and Data Analysis*. Chapman and Hall, 1986.

Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Petr Vaněk, Jan Mandel, and Marian Brezina. Algebraic multigrid by smoothed aggregation for second and fourth order elliptic problems. *Computing*, 56(3):179–196, 1996. doi: 10.1007/BF02238511.

Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C. J. Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental algorithms for scientific computing in Python. *Nature Methods*, 17:261–272, 2020. doi: 10.1038/s41592-019-0686-2.

Sifan Wang, Jacob Seidman, Shyam Sankaran, Hanwen Wang, George Pappas, and Paris Perdikaris. CViT: Continuous vision transformer for operator learning. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/3bf4b55960aaa23553cd2a6bdc6e1b57-Abstract-Conference.html.

Tian Wang and Chuang Wang. Latent neural operator for solving forward and inverse PDE problems. In *Advances in Neural Information Processing Systems*, 2024.

Haixu Wu, Huakun Luo, Haowen Wang, Jianmin Wang, and Mingsheng Long. Transolver: A fast transformer solver for pdes on general geometries. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 53681–53705, 2024a.

Wenxuan Wu, Zhongang Qi, and Li Fuxin. Pointconv: Deep convolutional networks on 3d point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9621–9630, 2019.

Xiaoyang Wu, Li Jiang, Peng-Shuai Wang, Zhijian Liu, Xihui Liu, Yu Qiao, Wanli Ouyang, Tong He, and Hengshuang Zhao. Point transformer v3: Simpler, faster, stronger. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4840–4851, 2024b.

Xumin Yu, Lulu Tang, Yongming Rao, Tiejun Huang, Jie Zhou, and Jiwen Lu. Point-BERT: Pre-training 3d point cloud transformers with masked point modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19313–19322, 2022.

APPENDIX

TABLE OF CONTENTS

A Input Representations and View Construction **14**

B Benchmark Physics, Targets, and Budgets **15**

C Training and Architecture Configurations **18**

D Accuracy Across Systems and Architectures **19**

E Prediction Stability Across Representations **21**

F Controlled Ablations of Views and Objectives **23**

G Case-Level Uncertainty and Output-Domain Error **26**

H Training-Seed Replication **27**

I Geometric Fidelity of Remeshing **29**

J Geometry, Representations, and Field Visualizations **30**

A INPUT REPRESENTATIONS AND VIEW CONSTRUCTION

The reference DeAL configuration estimates local concentration with $k = 16$ nearest neighbors and samples two fixed-size views without replacement. The Sine-x, Sine-y, Feature-aware, QEM, and Voxel-grid representations define unseen conditions. Within a Base–DeAL comparison, the physical case, surface and volume targets, query locations, target normalization, architecture, encoder-point count, and query count remain fixed.

Numerical view construction. Before KDE, coordinates are normalized by fixed training-set coordinate bounds of the surface. The applicable normalization rule is fixed across paired views of a case. The k-nearest-neighbor query excludes the center point. Distances, bandwidths, and kernels are evaluated in float64, and the resulting log-density is cached in float16. At sampling time it is promoted to float32, transformed according to Eq. (7), stabilized by subtracting the largest log-weight, and clamped below at 10^{-12}. Sampling is implemented in PyTorch (Paszke et al., 2019) with `torch.multinomial` without replacement. The Sine-x and Sine-y distributions use the floor $\epsilon = 10^{-2}$ after normalizing the selected coordinate to its source-cloud range. These choices avoid an additive density offset, keep the cache compact, and preserve the continuous interpretation of β.

Source clouds, encoder views, and remeshed inputs. We retain each case's available source cloud before selecting encoder views. For DrivAerML, 1,048,576 native surface points are sampled uniformly at random to limit storage. The heat-exchange point cloud contains 520,183–576,342 points. Their encoder views contain 131,072 and 65,536 points, respectively. This accounts for 12.5% on DrivAerML and 11.4–12.6% on the heat-exchange. SHIFT-Pump source surfaces contain 240,995–330,671 points per case and C-core surfaces contain 160,074–286,400 points; their 16,384-

and 32,768-point encoder views therefore occupy 5.0–6.8% and 11.4–20.5% of their respective sources. Each native view is sampled without replacement.

Source-cloud counts need not equal native mesh-vertex counts. Remeshing operates on the native triangular mesh, and a 5x or 10x label targets one-fifth or one-tenth of its original triangle count, rather than a prescribed vertex count. Table 1 reports the actual vertex ranges for the evaluated cohorts. Car and pump remeshes contain at least their respective encoder budgets and are sampled uniformly without replacement. Heat-exchange and C-core remeshes are sampled uniformly from remesh vertices, with replacement when their vertex counts fall below their respective encoder budgets of 65,536 and 32,768 points. In every case, the physical queries and targets remain unchanged.

Table 1: **Native and remeshed vertex counts.** Ranges pool the held-out cases and remeshing methods used for each system. The 5x and 10x labels denote target triangle-count reductions, not vertex-count or encoder-budget reductions.

Quantity	Car aerodynamics	Pump hydrodynamics	Heat exchange	C-core magnetostatics
Encoder budget	131,072	16,384	65,536	32,768
Native vertices	7.85–8.76M	240–330k	520–576k	153–281k
5x vertices	1.57–1.80M	46–63k	101–112k	32–59k
10x vertices	785–890k	22–30k	59–72k	16–28k
Remesh vertex sampling	Uniform, without replacement	Uniform, without replacement	Uniform, replacement if needed	Uniform, replacement if needed

Matched control distributions. The downsampling control uses a uniform secondary view with one quarter of the primary budget: 131,072 to 32,768 points for car, 16,384 to 4,096 for pump, 65,536 to 16,384 for heat exchange, and 32,768 to 8,192 for C-core. Gaussian-ball and box controls first draw a uniform primary-budget cloud, then select its center uniformly. Both use a scale equal to 5% of the largest source-cloud extent. The Gaussian control removes each point independently with probability 0.33 at one scale radius and a squared-exponential decay with distance; the box control removes points inside an axis-aligned box with side length twice that scale.

Attention normalization and surface partitions of unity. SMART's cross-attention weights sum to one over the keys available in a given point cloud. This gives a discrete convex combination, but not a fixed partition-of-unity basis on Γ: the support, keys, values, and softmax denominator all change with the sampled cloud X. Consequently, row-wise normalization does not remove the density ρ from Eq. (5) and does not by itself imply Eq. (3).

B Benchmark Physics, Targets, and Budgets

Data and budgets. For every task, encoder, surface-query, and volume-query budgets are held fixed within a paired comparison, and evaluation uses the held-out test subset. DrivAerML uses 131,072 encoder points and 65,536 queries in each domain. SHIFT-Pump uses 16,384 encoder points and 65,536 queries in each domain. The heat exchanger uses 65,536 encoder points and 32,768 queries in each domain. C-core magnetostatics uses a 32,768-point encoder budget and 65,536 surface and volume queries.

Target channels, normalization, and supervised loss. DrivAerML predicts surface pressure and three wall-shear components, together with volume pressure and three velocity components. SHIFT-Pump predicts surface pressure, three velocity components, and three wall-shear components, together with volume pressure and three velocity components. Heat exchange predicts scalar outward heat flux on the surface and scalar temperature in the volume. C-core predicts the three Cartesian components of magnetic flux density $\mathbf{B}$ in both domains. Vector quantities are represented by Cartesian components rather than magnitudes. For every output domain and channel, we compute one empirical mean and sample standard deviation from all target samples in the training split only, standardize targets with these fixed values, and use the same statistics for Base and DeAL.

For a domain d with channels c and query locations j, the branch discrepancy in Eq. (8) is

$$L_d = \frac{1}{C_d} \sum_{c=1}^{C_d} \sqrt{\frac{\sum_j (\widehat{Y}^d_{j,c} - Y^d_{j,c})^2}{\max\Big(\sum_j (Y^d_{j,c})^2, 10^{-5}\Big)}}, \qquad \ell = L_s + L_v, \tag{14}$$

where all quantities in this expression are standardized targets and predictions. Surface and volume discrepancies are summed during training; this training loss is distinct from the equal-domain, channelwise physical-space evaluation metric in Eq. (12).

We randomly partition cases once per dataset into disjoint 80%/10%/10% training, validation, and test subsets. The resulting train/validation/test counts are 387/48/49 for DrivAerML (484 cases), 1,120/140/140 for SHIFT-Pump (1,400 cases, uniform random pick from total), 230/29/29 for heat exchange (288 cases), and 220/27/28 for C-core magnetostatics (275 cases). The full SHIFT-Pump was not used due to storage restrictions. The test subset is fixed before model selection and is used only for the reported representation-shift evaluations.

DrivAerML. The released cases vary the notchback road-car geometry under fixed incompressible external-flow conditions. They use a $38.889\ \mathrm{m\,s^{-1}}$ freestream in a large open-road domain, with static wheels and sealed cooling inlets. The time-averaged surface and volume fields come from OpenFOAM v2212 finite-volume simulations using a hybrid RANS–LES turbulence model (Ashton et al., 2024).

SHIFT-Pump. We use the immutable dataset revision `8a332ffb079cbbc10a521112b8bcbbef0a03cf51`. Its provider card documents water as the working fluid, a rotating impeller, a stationary volute, and a sliding interface (Luminary Cloud, 2026). The corresponding low-Mach flow is summarized by the rotating-frame continuity and momentum balances

$$\nabla\cdot\mathbf{u} = 0, \tag{15}$$

$$\rho_f(\mathbf{u}\cdot\nabla)\mathbf{u} = -\nabla p + \nabla\cdot\boldsymbol{\tau}_{\text{eff}} + \mathbf{f}_\Omega, \tag{16}$$

where ρ_f is fluid density, $\boldsymbol{\tau}_{\text{eff}}$ collects viscous and any modeled stresses, and $\mathbf{f}_\Omega$ contains Coriolis and centrifugal terms in the impeller frame. We use this general form because the available dataset card does not identify a specific turbulence closure. It specifies a total-pressure inlet, an outlet mass-flow target, and rough impeller and volute walls. To distinguish cases with the same realized geometry at different operating conditions, we condition SHIFT-Pump models on $Q_{\text{op}} = \texttt{flow_rate} \times \texttt{flow_rate_op_condition}$. We learn the released surface and volume fields rather than resolving the provider's CFD system.

Heat exchange. This benchmark is a deterministic steady nonlinear partial differential equation (PDE) problem for the dimensionless temperature T on a case-specific solid domain Ω with channel-wall boundary Γ_h and exterior boundary Γ_e:

$$-\nabla\cdot\big[(1+aT^2)\nabla T\big] = 0 \qquad \text{in } \Omega, \tag{17}$$

$$T = 1 \qquad \text{on } \Gamma_h, \tag{18}$$

$$-(1+aT^2)\nabla T\cdot n = \mathrm{Bi}\,T + R\left[(T+\tau)^4 - \tau^4\right] \qquad \text{on } \Gamma_e. \tag{19}$$

All cases use the same dimensionless constants $a = 1.8$, $\mathrm{Bi} = 0.35$, $R = 0.07$, and $\tau = 1.5$. Here a controls the temperature-dependent conductivity, Bi the convective exchange, R the radiative exchange, and τ the ambient-temperature offset in the radiative term; n is the outward unit normal on the relevant boundary. The fixed channel-wall temperature supplies the heat source. All geometric variation is encoded in the surface itself. The reported surface field is the outward heat flux $-(1+aT^2)\nabla T\cdot n$, and the volume field is T. This makes the target a deterministic function of geometry under fixed material and boundary constants, rather than a hidden-variable prediction problem.

Heat-exchanger geometry generator. Each geometry is an extruded solid with height sampled uniformly from 1.25–1.75, thickness from 0.20–0.32, and independently perturbed left and right profiles. Each side is wavy on both, left-only, or right-only with equal probability; the two wave amplitudes are sampled from 0.025–0.075 and 0.010–0.045, their frequencies are integer-valued

from 1–3 and 3–6, and their phases are uniform on a full cycle. We request one to five through-channels uniformly, assign circular channels with probability 0.55 and rectangular channels otherwise, and use rejection sampling to enforce a 0.035 clearance from the exterior and between channels. Circular radii are 0.055–0.115; rectangular half-widths and half-heights are 0.050–0.115 and 0.050–0.125. A candidate is retained only when its minimum solid width is at least 0.55 and at least one nonoverlapping channel is placed; if a later requested channel cannot be placed, the valid earlier placements are retained.

Heat-exchanger discretization and verification. We construct each solid with Gmsh 4.15.2 (Geuzaine & Remacle, 2009) and generate an unstructured, curvature- and channel-proximity-refined linear tetrahedral mesh (3D algorithm 10 with Netgen optimization (Schöberl, 1997)). The PDE is discretized with first-order tetrahedral finite elements in scikit-fem 12.0.2 (Gustafsson & McBain, 2020). A damped Picard iteration updates both the temperature-dependent conductivity and radiative boundary coefficient; each symmetric positive-definite linear subproblem is solved with smoothed-aggregation algebraic multigrid (AMG)-preconditioned conjugate gradients (Vaněk et al., 1996) (relative tolerance 10^{-10}, at most 20,000 iterations), with a sparse direct fallback. A case is accepted only when the linear relative residual is below 2×10^{-8}, the relative nonlinear update is below 2×10^{-7}, and the channel Dirichlet error is below 10^{-12}. Across the saved corpus, the largest recorded linear residual is 9.9×10^{-11} and the largest nonlinear update is 1.6×10^{-7}. Every case converges in a maximum of 14 Picard iterations. We also compare independently regenerated meshes with a $1.35\times$ coarser target size (about 44% of the native tetrahedra) at shared interior locations. The temperature difference is 3.19% on average (3.65% maximum) in relative ℓ_2, while the integrated exterior heat-flow difference is 1.38% on average (2.23% maximum). All sampled locations are shared by the two meshes.

C-core magnetostatics. The benchmark fixes winding excitation and material permeability while varying the connected ferromagnetic-core geometry, including the yoke profile, aperture configuration, gap, and cross-section. We solve a reduced scalar-potential correction to an analytic impressed winding field on a conformal core–air domain, and retain only solid-core surface and volume samples for learning. The target is the magnetic flux-density vector $\mathbf{B} = \mu_0\mu_r(\mathbf{H}_{\mathrm{imp}} - \nabla\phi)$; the surrounding air region is used by the solver but is not a surrogate input or prediction target. Thus the magnetic field is a deterministic geometry-to-field map under fixed excitation and material settings.

C-core geometry and magnetic specification. We perturb eight validated C-core templates spanning rectangular, circular, and elliptical yokes; unified or dual apertures; parallel or focused poles; and rectangular or circular cross-sections. Per case, width and height are scaled by 0.96–1.04, the two window ratios receive independent plus-or-minus 0.015 perturbations and are clipped to 0.43–0.59 and 0.43–0.65, and the core center is shifted by plus or minus 0.002. Parallel-pole gaps are scaled by 0.78–1.22 and clipped to 0.0045–0.026. Focused-pole inner and tip gaps are scaled by 0.88–1.12 and 0.80–1.20, clipped to 0.020–0.035 and 0.0035–0.016, with the tip gap at most 72% of the inner gap. The computational domain is the air box from $(-0.150, -0.135, -0.088)$ to $(0.180, 0.135, 0.088)$. The impressed field is the superposition of ten circular filaments of radius 0.0265 centered on the y-axis at $x = -0.0665$, equally spaced from -0.035 to 0.035, each carrying 2 A. The coil therefore has 20 A-turn magnetomotive force and is excluded from the solve mesh after defining the air–coil interface. We solve $\nabla\cdot[\mu_r(\nabla\phi - \mathbf{H}_{\mathrm{imp}})] = 0$ with $\mu_r = 300$ in the core and $\mu_r = 1$ in air, imposing $\phi = 0$ on the outer air-box boundary to fix the scalar-potential gauge.

C-core discretization and verification. Gmsh 4.15.2 produces a conformal, unstructured linear-tetrahedral core–air mesh with curvature- and winding-proximity refinement, followed by Netgen optimization. The core has fixed relative permeability $\mu_r = 300$ and the analytic winding has fixed magnetomotive force 20 A-turn. We solve the sparse first-order finite-element scalar-potential system using sparse direct factorization in SciPy 1.16.0 (Virtanen et al., 2020) and require a relative algebraic residual below 10^{-10}. Across the completed corpus, the maximum accepted residual is 7.4×10^{-12}. The 95th percentile of the mirror-symmetry audit is 0.70%, below its 8% acceptance limit. The solid surface is watertight in every accepted case. A two-grid check on representative unified-window cores gives 2.1–2.5% relative ℓ_2 difference in $\mathbf{B}$ and 1.7–2.0% in $\|\mathbf{B}\|$ at shared interior locations.

Optimization and loss settings. Each Base–DeAL pair shares the backbone, optimizer family, normalization, query budgets, and physical training cases. The reference DeAL objective uses two supervised-view terms and one prediction-consistency term with fixed weights $(0.2, 0.2, 0.6)$; the weighting ablation changes only this aggregation rule. The consistency discrepancy is symmetric stop-gradient Smooth-ℓ_1 with unit coefficient and transition parameter 0.1. Architecture-study percentages are derived from the means of Eq. (12).

C Training and Architecture Configurations

Each Base–DeAL pair uses independently initialized copies of the same backbone. We use a batch size of two physical cases and single-precision floating-point (FP32) model arithmetic throughout. Base is optimized for 300 epochs with one input view per case. DeAL is optimized for 150 epochs, with two independently sampled views per case in every iteration. The two branch losses are aggregated before one optimizer step, so the schedules match in view-level supervised branches rather than in optimizer-step count. Table 2 specifies the shared optimizer, schedule, and hardware.

Tables 3–5 report measured per-device compute time, backbone configurations, and point budgets. DeAL preserves each encoder and query budget, but instantiates the encoder budget as two independently sampled views. Parameter counts are rounded to 0.1M because task-specific output heads and Pump-conditioning vary it slightly.

Table 2: **Shared optimization and implementation settings.** All Base–DeAL pairs use these settings unless a dataset-specific architecture configuration is stated below.

Item	Specification
Initialization	Base and DeAL use independent random initialization copies of the same backbone.
Training schedule	Base: 300 epochs with one view per case. DeAL: 150 epochs with two independently sampled views per case. Branch losses are aggregated before one optimizer step.
Numerics	Batch size two with FP32 model arithmetic and no gradient clipping.
Optimizer	AdamW (Loshchilov & Hutter, 2019) with initial learning rate 2×10^{-4} and weight decay 10^{-6}. Bias, normalization, and position-related parameters have no weight decay.
Learning-rate schedule	Cosine annealing (Loshchilov & Hutter, 2017) stepped after every optimizer update over the complete training run.
Hardware	NVIDIA RTX A5000 GPUs with 24 GB memory. Profiled node: two Intel Xeon Gold 6242R processors (40 physical CPU cores, 80 hardware threads) and 754 GB RAM.

Table 3: **Forward–backward time per optimizer step (s).** Median of 25 per-device measurements after 10 warm-up updates; mean DeAL/Base time ratio: 2.046.

System	Training	SMART	AB-UPT	Geo-FNO	PointNet++	PointGNN	LNO	MSPT	Transolver++	PTv3
Car	Base	0.389	0.522	0.088	0.122	0.700	0.153	0.393	0.307	0.378
	DeAL	0.818	1.066	0.177	0.250	1.437	0.299	0.824	0.621	0.748
Pump	Base	0.264	0.345	0.067	0.115	0.381	0.073	0.293	0.235	0.256
	DeAL	0.538	0.692	0.135	0.238	0.778	0.145	0.613	0.506	0.498
Heat	Base	0.167	0.341	0.060	0.084	0.360	0.144	0.252	0.208	0.285
	DeAL	0.343	0.701	0.122	0.171	0.737	0.276	0.534	0.451	0.560
C-core	Base	0.236	0.274	0.064	0.098	0.422	0.071	0.293	0.234	0.430
	DeAL	0.493	0.566	0.131	0.202	0.865	0.142	0.612	0.505	0.872

PointNet++ uses $2{,}048/512$ set-abstraction centroids for car, pump, and heat, and $512/128$ for C-core.

Table 4: **Shared backbone configurations.** SHIFT-Pump supplies one case-level scalar Q_{op} (`parameter_channels` $=$ 1) to Base and both DeAL views; other systems use zero conditioning channels. Rows specify its projection and injection site.

Backbone	Shared architecture	Pump conditioning	Params. (M)
SMART	$d = 256$, six encoder–decoder cross-attention blocks, and position scale 100.	$1 \to 128 \to 8d$ gives scale/shift for each encoder/decoder $4d$ MLP hidden state.	7.1
AB-UPT	$d = 192$, six heads, and geometry/interleaved/self/decoder depths $2/2/6/2$.	$1 \to d$ additions to geometry/anchor/query tokens and $1 \to 128 \to 8d$ scale/shift of block MLPs.	10.7
Geo-FNO	Width 64, latent resolution 24, four Fourier blocks with five modes, and a 128-width deformation network with eight Fourier bands.	$1 \to 128$ coordinate-deformation code and separate $1 \to 64$ additions to grid and query features.	24.4
PointNet++ SSG	$d = 256$, set-abstraction caps $64/128$, and three-neighbor query interpolation.	Separate $1 \to 256$ projections added after global geometry pooling and to query tokens.	2.9
PointGNN	Four radius-graph blocks with $d = 256$, node width 64, edge width 128, auto-centering, three voxel-pooling stages, 16 graph neighbors, eight query neighbors, and three query blocks. The radii are $0.035/0.08/0.16/0.24$.	Separate $1 \to 256$ projections added to pooled graph latent and query features.	2.0
LNO	$d = 256$, 256 modes, eight operator blocks, eight heads, two decoder layers, and Galerkin attention.	Separate $1 \to 256$ projections added to branch-point and query features.	5.1
MSPT	$d = 192$, 128 value tokens, one query token, four blocks, four heads, and ball-tree pooling.	One $1 \to 192 \to 192$ GELU MLP added to geometry and query tokens before attention.	2.2
Transolver++	$d = 256$, 32 learned slices, four layers, eight heads, Gumbel slice assignment, and multilayer-perceptron (MLP) ratio two.	One $1 \to 256 \to 256$ GELU MLP added to geometry and query tokens before slice assignment.	1.7
Point Transformer V3	Encoder depths $1/1/2/4/1$, channels $24/48/96/192/256$, and decoder depths $1/1/1/1$.	Separate $1 \to 256$ projections added to pooled geometry and query tokens before cross-attention.	11.0

Table 5: **Dataset-specific input and hierarchy budgets.** SMART entries give sampled points/geometry latents, and AB-UPT entries give supernodes/surface–volume anchors.

System	Encoder input	Surface queries	Volume queries	SMART	AB-UPT	PTv3 local configuration
Car	131,072	65,536	65,536	16,384/4,096	2,048/1,024/1,024	1,024-point patches.
Pump	16,384	65,536	65,536	2,048/512	1,024/512/512	512-point patches.
Heat	65,536	32,768	32,768	8,192/2,048	2,048/1,024/1,024	1,024-point patches.
C-core	32,768	65,536	65,536	4,096/1,024	512/256/256	512-point patches.

D Accuracy Across Systems and Architectures

Table 6 applies the same paired metric and aggregation to all four systems and nine architectures. Base is trained with conventional single-view supervision using only $\beta = 0$ inputs. DeAL instead samples $\beta \sim \mathcal{U}[0, 1]$ for its training views. Consequently, the label "unshifted input" is not applicable to DeAL because it has no unique reference density. For completeness, the first two columns evaluate both models at the Base reference density $\beta = 0$. The remaining columns measure test shifts from axis redistributions and independently generated remeshes. Table 7 expands the DrivAerML results by representation condition, while Figure 5 shows the paired percentage reduction for every task–architecture combination.

Uncertainty convention. Across the main-text bar charts with whiskers (Figures 1, 2, 3(e), and 4), each whisker denotes the case-level standard deviation of the plotted mean relative-ℓ_2 field error. The related tables below report the corresponding numerical means and standard deviations.

Table 6: **Error at $\beta = 0$ and under test shifts across systems and architectures.** The $\beta = 0$ columns provide a common reference input for completeness; test-shift values average axis redistributions and mean 5x and 10x remeshing. Entries are mean $\pm$ case-level standard deviation of relative-ℓ_2 field error; reduction compares paired test-shift means.

Physical system	Architecture	$\beta = 0$ Base	$\beta = 0$ DeAL	Test shifts Base	Test shifts DeAL	Reduction
Car aerodynamics						
	SMART	$.074 \pm .006$	$.071 \pm .007$	$.381 \pm .018$	$\mathbf{.188 \pm .012}$	$\downarrow 50.7\%$
	AB-UPT	$.243 \pm .012$	$.195 \pm .009$	$.327 \pm .021$	$\mathbf{.234 \pm .018}$	$\downarrow 28.4\%$
	Geo-FNO	$.402 \pm .013$	$.387 \pm .014$	$1.025 \pm .014$	$\mathbf{.501 \pm .015}$	$\downarrow 51.1\%$
	PointNet++ SSG	$.358 \pm .026$	$.329 \pm .025$	$.744 \pm .015$	$\mathbf{.422 \pm .028}$	$\downarrow 43.3\%$
	PointGNN	$.292 \pm .019$	$.287 \pm .010$	$.497 \pm .019$	$\mathbf{.304 \pm .011}$	$\downarrow 38.9\%$
	LNO	$.443 \pm .012$	$.423 \pm .019$	$.992 \pm .040$	$\mathbf{.495 \pm .014}$	$\downarrow 50.1\%$
	MSPT	$.193 \pm .010$	$.182 \pm .008$	$.313 \pm .019$	$\mathbf{.199 \pm .009}$	$\downarrow 36.2\%$
	Transolver++	$.166 \pm .008$	$.166 \pm .008$	$.283 \pm .010$	$\mathbf{.180 \pm .008}$	$\downarrow 36.1\%$
	Point Transformer V3	$.472 \pm .024$	$.429 \pm .023$	$.567 \pm .016$	$\mathbf{.455 \pm .015}$	$\downarrow 19.7\%$
Pump hydrodynamics						
	SMART	$.139 \pm .004$	$.115 \pm .006$	$.297 \pm .014$	$\mathbf{.173 \pm .015}$	$\downarrow 41.9\%$
	AB-UPT	$.214 \pm .010$	$.189 \pm .006$	$.239 \pm .010$	$\mathbf{.170 \pm .005}$	$\downarrow 28.7\%$
	Geo-FNO	$.301 \pm .015$	$.271 \pm .016$	$.570 \pm .014$	$\mathbf{.349 \pm .011}$	$\downarrow 38.7\%$
	PointNet++ SSG	$.325 \pm .006$	$.262 \pm .008$	$.438 \pm .024$	$\mathbf{.277 \pm .013}$	$\downarrow 36.7\%$
	PointGNN	$.191 \pm .007$	$.198 \pm .004$	$.352 \pm .006$	$\mathbf{.208 \pm .004}$	$\downarrow 41.1\%$
	LNO	$.384 \pm .009$	$.363 \pm .012$	$.399 \pm .012$	$\mathbf{.308 \pm .012}$	$\downarrow 22.8\%$
	MSPT	$.214 \pm .005$	$.219 \pm .007$	$.316 \pm .006$	$\mathbf{.218 \pm .007}$	$\downarrow 31.0\%$
	Transolver++	$.183 \pm .007$	$.165 \pm .007$	$.192 \pm .008$	$\mathbf{.136 \pm .007}$	$\downarrow 29.3\%$
	Point Transformer V3	$.356 \pm .017$	$.340 \pm .028$	$.506 \pm .018$	$\mathbf{.370 \pm .013}$	$\downarrow 26.9\%$
Heat exchange						
	SMART	$.113 \pm .016$	$.117 \pm .024$	$.467 \pm .075$	$\mathbf{.248 \pm .040}$	$\downarrow 47.0\%$
	AB-UPT	$.108 \pm .017$	$.107 \pm .017$	$.345 \pm .107$	$\mathbf{.144 \pm .023}$	$\downarrow 58.2\%$
	Geo-FNO	$.171 \pm .048$	$.167 \pm .050$	$.663 \pm .149$	$\mathbf{.434 \pm .152}$	$\downarrow 34.6\%$
	PointNet++ SSG	$.365 \pm .243$	$.365 \pm .166$	$.622 \pm .132$	$\mathbf{.445 \pm .145}$	$\downarrow 28.4\%$
	PointGNN	$.312 \pm .113$	$.317 \pm .077$	$.567 \pm .066$	$\mathbf{.336 \pm .050}$	$\downarrow 40.8\%$
	LNO	$.154 \pm .019$	$.143 \pm .024$	$.971 \pm .117$	$\mathbf{.469 \pm .083}$	$\downarrow 51.7\%$
	MSPT	$.109 \pm .021$	$.107 \pm .022$	$.422 \pm .095$	$\mathbf{.155 \pm .060}$	$\downarrow 63.2\%$
	Transolver++	$.158 \pm .036$	$.138 \pm .031$	$.698 \pm .107$	$\mathbf{.284 \pm .123}$	$\downarrow 59.4\%$
	Point Transformer V3	$.153 \pm .013$	$.169 \pm .016$	$.641 \pm .174$	$\mathbf{.294 \pm .055}$	$\downarrow 54.1\%$
C-core magnetostatics						
	SMART	$.062 \pm .019$	$.063 \pm .020$	$.190 \pm .058$	$\mathbf{.102 \pm .034}$	$\downarrow 46.3\%$
	AB-UPT	$.099 \pm .010$	$.086 \pm .012$	$.154 \pm .023$	$\mathbf{.105 \pm .007}$	$\downarrow 31.7\%$
	Geo-FNO	$.108 \pm .008$	$.105 \pm .007$	$.764 \pm .124$	$\mathbf{.361 \pm .059}$	$\downarrow 52.7\%$
	PointNet++ SSG	$.124 \pm .008$	$.133 \pm .012$	$.525 \pm .071$	$\mathbf{.205 \pm .072}$	$\downarrow 60.9\%$
	PointGNN	$.180 \pm .006$	$.174 \pm .005$	$.379 \pm .115$	$\mathbf{.191 \pm .058}$	$\downarrow 49.8\%$
	LNO	$.165 \pm .003$	$.145 \pm .004$	$1.438 \pm .099$	$\mathbf{.441 \pm .070}$	$\downarrow 69.3\%$
	MSPT	$.107 \pm .011$	$.080 \pm .014$	$.225 \pm .067$	$\mathbf{.099 \pm .027}$	$\downarrow 55.9\%$
	Transolver++	$.110 \pm .005$	$.101 \pm .006$	$.188 \pm .020$	$\mathbf{.102 \pm .006}$	$\downarrow 45.9\%$
	Point Transformer V3	$.150 \pm .015$	$.146 \pm .014$	$.506 \pm .043$	$\mathbf{.374 \pm .011}$	$\downarrow 26.1\%$

Table 7: **Cross-architecture DrivAerML results on unseen representations.** Each architecture is a two-row block: Base is listed first and DeAL second. DeAL cells show mean ± case-level standard deviation with relative reduction (↓) underneath. Remeshing columns average Feature-aware, QEM, and Voxel-grid inputs. The SMART block corresponds to Figure 2(a). Lower is better.

Architecture	Training	Sine-x	Sine-y	Remesh 5x	Remesh 10x
SMART	Base	.217 ± .036	.526 ± .028	.379 ± .019	.401 ± .016
	DeAL	**.126 ± .022**	**.077 ± .007**	**.265 ± .022**	**.284 ± .016**
	vs. Base	↓ 42.0%	↓ 85.4%	↓ 30.1%	↓ 29.3%
AB-UPT	Base	.279 ± .014	.292 ± .032	.357 ± .025	.379 ± .026
	DeAL	**.200 ± .010**	**.199 ± .013**	**.262 ± .027**	**.274 ± .032**
	vs. Base	↓ 28.2%	↓ 32.0%	↓ 26.5%	↓ 27.6%
Geo-FNO	Base	.791 ± .043	1.083 ± .033	1.081 ± .021	1.144 ± .025
	DeAL	**.399 ± .019**	**.391 ± .017**	**.599 ± .014**	**.614 ± .016**
	vs. Base	↓ 49.5%	↓ 63.9%	↓ 44.6%	↓ 46.3%
PointNet++ SSG	Base	.547 ± .054	.914 ± .006	.760 ± .016	.755 ± .020
	DeAL	**.343 ± .029**	**.349 ± .033**	**.529 ± .042**	**.468 ± .031**
	vs. Base	↓ 37.3%	↓ 61.8%	↓ 30.4%	↓ 38.1%
PointGNN	Base	.466 ± .018	.485 ± .030	.513 ± .014	.523 ± .017
	DeAL	**.282 ± .010**	**.294 ± .014**	**.317 ± .010**	**.322 ± .010**
	vs. Base	↓ 39.5%	↓ 39.4%	↓ 38.2%	↓ 38.4%
LNO	Base	.889 ± .088	.986 ± .033	1.040 ± .047	1.052 ± .050
	DeAL	**.431 ± .016**	**.425 ± .013**	**.562 ± .015**	**.563 ± .019**
	vs. Base	↓ 51.4%	↓ 56.9%	↓ 46.0%	↓ 46.5%
MSPT	Base	.244 ± .011	.419 ± .029	.286 ± .019	.302 ± .021
	DeAL	**.191 ± .008**	**.206 ± .014**	**.199 ± .008**	**.202 ± .008**
	vs. Base	↓ 21.8%	↓ 50.9%	↓ 30.4%	↓ 33.1%
Transolver++	Base	.250 ± .012	.341 ± .022	.261 ± .007	.278 ± .008
	DeAL	**.168 ± .009**	**.168 ± .008**	**.191 ± .007**	**.195 ± .008**
	vs. Base	↓ 32.7%	↓ 50.8%	↓ 26.8%	↓ 30.1%
Point Transformer V3	Base	.536 ± .028	.571 ± .022	.570 ± .014	.591 ± .014
	DeAL	**.435 ± .029**	**.459 ± .019**	**.460 ± .014**	**.467 ± .014**
	vs. Base	↓ 18.9%	↓ 19.6%	↓ 19.3%	↓ 20.9%

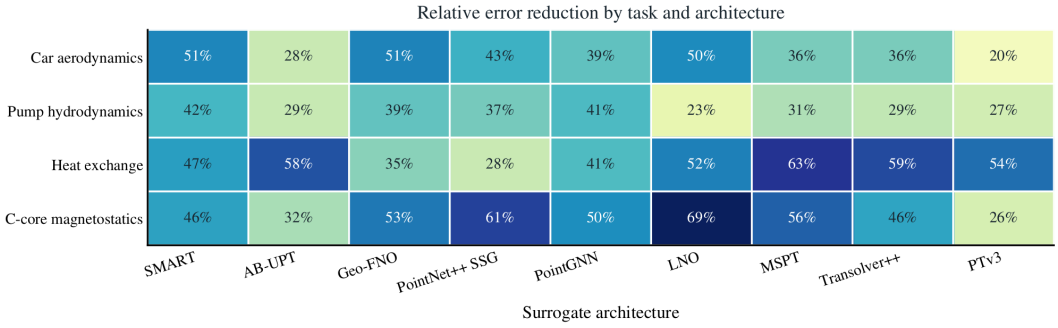


Figure 5: **Paired reduction by task and architecture.** Each cell gives the relative reduction in mean relative-ℓ_2 field error from the matched Base to DeAL under test shifts. Every task–architecture pair shows a positive reduction, demonstrating transfer across the evaluated backbones and physical systems. Table 6 reports the corresponding absolute means and case-level standard deviations.

E Prediction Stability Across Representations

The target-relative error in Eq. (12) assesses accuracy under each representation, but it does not directly measure how much two predictions change when only their numerical representations differ. We therefore evaluate direct prediction disagreement (DD) at the same physical query locations. For a reference representation X_0, an unseen representation X_r, and method m, we use

$$D_m(X_0, X_r) = \frac{1}{2} \sum_{d \in \{s,v\}} \frac{1}{C_d} \sum_{c=1}^{C_d} \frac{\left\| \widehat{Y}^d_{m,:,c}(X_0) - \widehat{Y}^d_{m,:,c}(X_r) \right\|_2}{\left\| Y^d_{:,c} \right\|_2}. \tag{20}$$

This uses the same domain weighting and channelwise physical normalization as Eq. (12), while replacing the target residual in the numerator with the difference between predictions. DD is thus distinct from target-relative error and directly evaluates the representation-stability objective in Eq. (3).

For each case, we compare the reference representation with Sine-x, Sine-y, and the method-averaged 5x and 10x remeshes at common queries, assigning equal weight to the four shift families before aggregating across the held-out test subset. Figure 6 and Table 8 report the resulting means for every matched task–architecture pair. DeAL has lower DD in all 36 pairs, with a macro-average paired reduction of 49.8%. The direction holds across point hierarchies, graph message passing, operator models, and attention-based encoders, providing direct evidence that the improvement under unseen shifts is accompanied by smaller representation-induced prediction changes.

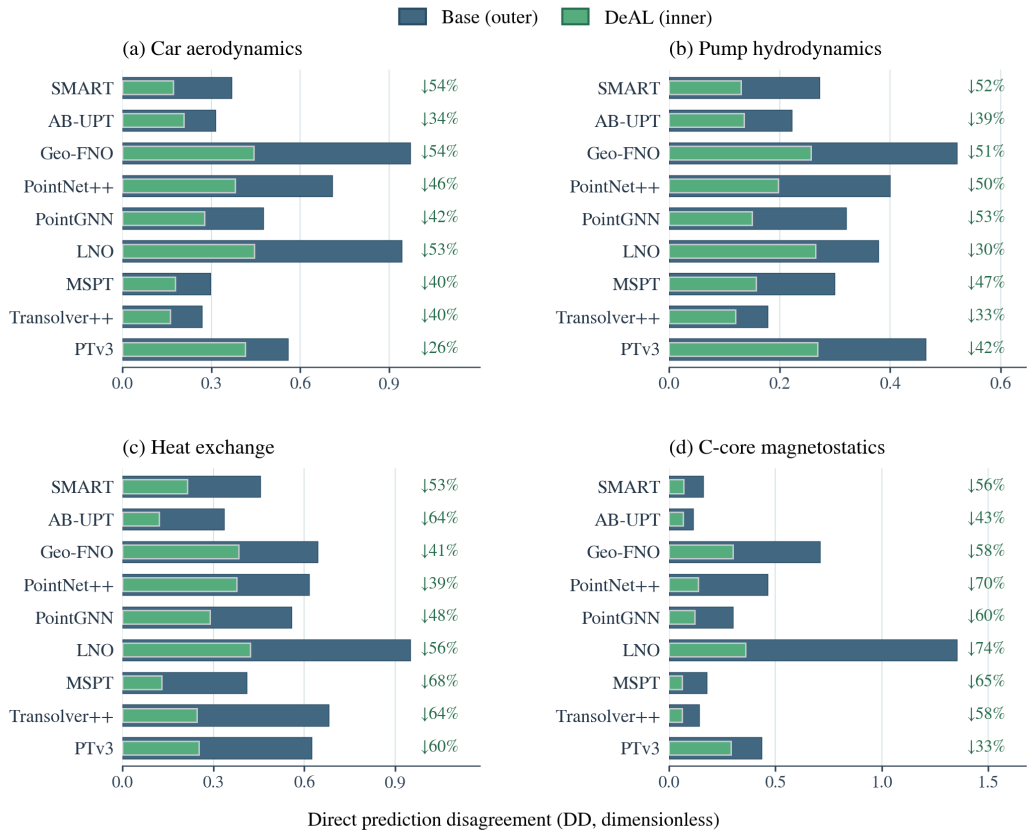


Figure 6: **Direct prediction disagreement across architectures and systems.** Outer blue bars show Base and inner green bars show DeAL mean DD over the test subset; percentage labels give rounded paired reductions. Each comparison evaluates a reference representation and held-out representation-equivalent inputs at common physical queries. DeAL lowers DD in every matched pair, showing that its shifted-input gains coincide with more stable predictions across equivalent representations. DD is a dimensionless channelwise ratio normalized by the physical target as in Eq. (20); it is separate from target-relative field error.

Table 8: **Direct prediction disagreement across systems and architectures.** Entries are mean DD across the held-out representation shifts defined above. Each reduction compares the matched Base and DeAL means. Lower DD indicates more similar predictions for representations of the same physical case.

Car aerodynamics

Architecture	Base DD	DeAL DD	Reduction
SMART	0.368	**0.171**	↓ 53.5%
AB-UPT	0.314	**0.208**	↓ 33.7%
Geo-FNO	0.972	**0.444**	↓ 54.4%
PointNet++	0.709	**0.380**	↓ 46.4%
PointGNN	0.475	**0.277**	↓ 41.6%
LNO	0.942	**0.445**	↓ 52.7%
MSPT	0.298	**0.178**	↓ 40.1%
Transolver++	0.269	**0.162**	↓ 39.7%
PTv3	0.558	**0.415**	↓ 25.7%

Pump hydrodynamics

Architecture	Base DD	DeAL DD	Reduction
SMART	0.272	**0.130**	↓ 52.2%
AB-UPT	0.222	**0.136**	↓ 38.6%
Geo-FNO	0.521	**0.257**	↓ 50.7%
PointNet++	0.400	**0.198**	↓ 50.5%
PointGNN	0.321	**0.150**	↓ 53.4%
LNO	0.379	**0.265**	↓ 30.1%
MSPT	0.300	**0.158**	↓ 47.3%
Transolver++	0.179	**0.120**	↓ 33.2%
PTv3	0.464	**0.269**	↓ 42.1%

Heat exchange

Architecture	Base DD	DeAL DD	Reduction
SMART	0.455	**0.215**	↓ 52.7%
AB-UPT	0.336	**0.122**	↓ 63.6%
Geo-FNO	0.645	**0.384**	↓ 40.6%
PointNet++	0.616	**0.378**	↓ 38.7%
PointGNN	0.559	**0.289**	↓ 48.3%
LNO	0.951	**0.422**	↓ 55.6%
MSPT	0.411	**0.130**	↓ 68.3%
Transolver++	0.681	**0.246**	↓ 63.9%
PTv3	0.625	**0.252**	↓ 59.7%

C-core magnetostatics

Architecture	Base DD	DeAL DD	Reduction
SMART	0.161	**0.070**	↓ 56.5%
AB-UPT	0.115	**0.066**	↓ 42.7%
Geo-FNO	0.710	**0.301**	↓ 57.6%
PointNet++	0.465	**0.138**	↓ 70.2%
PointGNN	0.302	**0.120**	↓ 60.4%
LNO	1.354	**0.359**	↓ 73.5%
MSPT	0.179	**0.063**	↓ 64.6%
Transolver++	0.143	**0.061**	↓ 57.6%
PTv3	0.435	**0.292**	↓ 32.8%

F Controlled Ablations of Views and Objectives

The matched view-generator controls in Table 9 hold the two-view supervision and prediction-consistency objective fixed while varying only view construction, isolating the contribution of inverse-density sampling from that of generic paired training. The following ablations then test the density range, density neighborhood, consistency term, and loss balancing separately.

Table 9: **Matched SMART view-generator controls across physical systems.** Entries are mean $\pm$ case-level standard deviation of relative-ℓ_2 field error under the same four shifted conditions. Every non-Base strategy retains two-view supervision and prediction consistency; only its view generator changes. Within each physical system, all methods use the same cases, queries, condition definitions, and physical-space metric. Lower is better.

Task	Shift	Base	Downsample	Gaussian ball	Box mask	DeAL
Car aerodynamics	Sine-x	.217 ± .036	.201 ± .026	.211 ± .033	.199 ± .030	**.126 ± .022**
	Sine-y	.526 ± .028	.332 ± .053	.394 ± .061	.296 ± .056	**.077 ± .007**
	Remesh 5x	.379 ± .019	.318 ± .016	.382 ± .016	.333 ± .013	**.265 ± .022**
	Remesh 10x	.401 ± .016	.315 ± .019	.368 ± .021	.318 ± .018	**.284 ± .016**
Pump hydrodynamics	Sine-x	.198 ± .019	.139 ± .007	.137 ± .006	.144 ± .007	**.116 ± .006**
	Sine-y	.169 ± .011	.130 ± .005	.146 ± .007	.148 ± .005	**.107 ± .006**
	Remesh 5x	.408 ± .024	.281 ± .023	.306 ± .028	.333 ± .027	**.235 ± .041**
	Remesh 10x	.413 ± .022	.279 ± .022	.305 ± .026	.332 ± .026	**.233 ± .022**
Heat exchange	Sine-x	.466 ± .206	.358 ± .139	.284 ± .044	.448 ± .196	**.125 ± .023**
	Sine-y	.429 ± .032	.382 ± .034	.389 ± .028	.418 ± .037	**.329 ± .014**
	Remesh 5x	.514 ± .107	.468 ± .092	.449 ± .081	.543 ± .125	**.281 ± .078**
	Remesh 10x	.461 ± .075	.406 ± .060	.387 ± .051	.480 ± .085	**.256 ± .085**
C-core magnetostatics	Sine-x	.341 ± .094	.206 ± .047	.215 ± .051	.217 ± .050	**.097 ± .013**
	Sine-y	.115 ± .042	.123 ± .021	.130 ± .021	.131 ± .017	**.066 ± .020**
	Remesh 5x	.142 ± .049	.134 ± .023	.152 ± .027	.145 ± .017	**.117 ± .047**
	Remesh 10x	.161 ± .068	.146 ± .031	.172 ± .040	.163 ± .031	**.128 ± .059**

For the spatial comparison in Figure 3(b–d), we select the lowest-error matched control independently for each system: Box mask for car aerodynamics, Downsampling for pump hydrodynamics and C-core magnetostatics, and Gaussian-ball masking for heat exchange.

Continuous density variation and fixed-intensity controls. In the reference DeAL protocol, we independently draw the two density exponents from $\mathcal{U}[0, 1]$. To distinguish this continuum from paired sampling alone and from a small set of fixed settings, we hold the SMART backbone, paired supervision, consistency term, point budget, and evaluation protocol fixed. The fixed $\beta = 0$ and $\beta = 1$ controls draw two independent views at their respective exponents. The discrete endpoint control places one view at $\beta = 0$ and the other at $\beta = 1$.

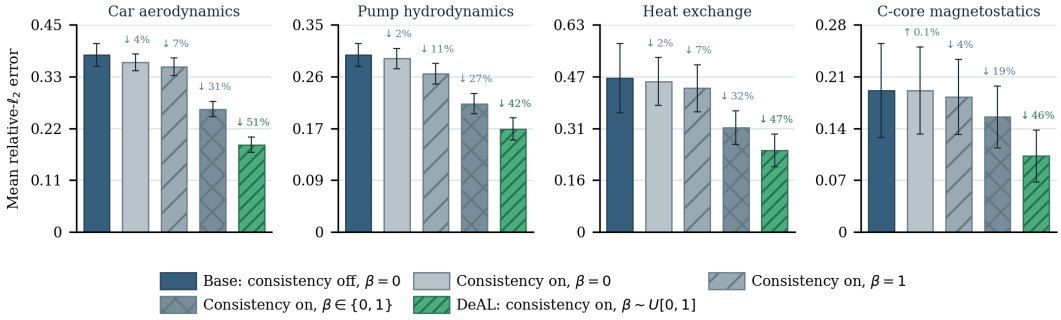


Figure 7: **Continuous-density controls across physical systems.** Base uses one supervised view at $\beta = 0$ without consistency. The next three bars use paired supervision and consistency with both views fixed at $\beta = 0$, both fixed at $\beta = 1$, or one view at each endpoint. DeAL independently draws each exponent from a continuous uniform distribution. Bars show mean relative-ℓ_2 error over held-out representation shifts, whiskers show mean condition-level case standard deviation, and labels give reductions from Base. Continuous inverse-density draws yield the lowest error in every system, exceeding the fixed-intensity and endpoint alternatives.

Table 10: **Continuous-density control ablation across physical systems.** Entries report mean $\pm$ mean condition-level case standard deviation of relative-ℓ_2 field error over held-out representation shifts. The five columns use the protocols shown in Figure 7. Lower is better.

Dataset	Base off, $\beta = 0$	Consistency on $\beta = 0$	Consistency on $\beta = 1$	Consistency on $\beta \in \{0, 1\}$	DeAL, consistency on $\beta \sim \mathcal{U}[0, 1]$
Car aerodynamics	$.381 \pm .025$	$.365 \pm .018$	$.355 \pm .019$	$.264 \pm .017$	$\mathbf{.188 \pm .017}$
Pump hydrodynamics	$.297 \pm .019$	$.291 \pm .017$	$.266 \pm .017$	$.216 \pm .017$	$\mathbf{.173 \pm .019}$
Heat exchange	$.467 \pm .105$	$.457 \pm .072$	$.437 \pm .071$	$.317 \pm .052$	$\mathbf{.248 \pm .050}$
C-core magnetostatics	$.190 \pm .063$	$.190 \pm .058$	$.181 \pm .050$	$.154 \pm .041$	$\mathbf{.102 \pm .035}$

Figure 7 and Table 10 show that paired views fixed at $\beta = 0$ and $\beta = 1$ yield mean reductions of 2.1% and 7.1% from Base across systems, respectively. The discrete endpoint pair reaches 27.2%, whereas independent continuous draws reduce error by 46.4% and are lowest in every system. Within this matched ablation, the gap indicates that the endpoint pair does not reproduce the benefit of sampling intermediate densities.

View-construction sensitivity. We fixed $\beta_{\max} = 1$ and $k = 16$ before the held-out-shift ablations and did not select either setting using shifted-test error. The endpoint $\beta_{\max} = 1$ spans uniform sampling at $\beta = 0$ through full inverse-density weighting, making it a natural reference range. We chose $k = 16$ as a moderate reference neighborhood, drawing on local grouping in PointNet++ (Qi et al., 2017) and the 16-point neighborhood used in Point Transformer V3 (Wu et al., 2024b). Their experience with local neighborhoods motivates this choice, although we use the neighbors to estimate source-point concentration rather than to aggregate features. Table 11 and Figure 8 report retrospective sensitivity analyses results. Increasing $\beta_{\max}$ improves aggregate error through the reference range, followed by a shallow plateau around 1–2. Increasing k improves remeshing error through $k = 32$, while the two sine conditions remain close across the tested neighborhoods.

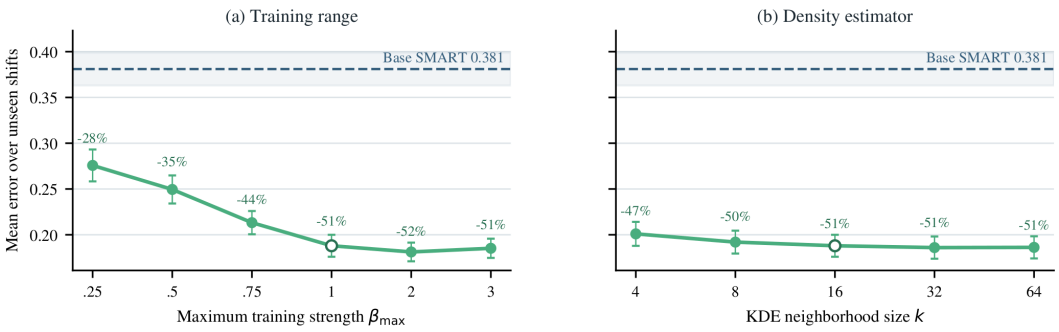


Figure 8: **Controlled DeAL studies on unseen representation shifts.** Each scalar gives equal weight to Sine-x, Sine-y, mean 5x remeshing, and mean 10x remeshing. (a) Training strength. (b) KDE neighborhood size. Dashed lines denote Base SMART, shaded bands and error bars show case-level standard deviation, and open markers identify the reference settings. Each point is labeled with its reduction from the labeled Base SMART reference. The preselected $\beta_{\max} = 1$ and $k = 16$ are marked to show sensitivity around the reference configuration, not to select it from test results.

Table 11: **DrivAerML SMART view-construction ablations.** (a) Upper endpoint of $\beta \sim \mathcal{U}[0, \beta_{\max}]$. (b) k-nearest-neighbor neighborhood size for the density estimator. Entries are relative-ℓ_2 field error on unseen representations. Base SMART and the reference settings $[0, 1]$, $k = 16$ reproduce Figure 2(a). Lower is better.

(a) Training range

Training range	Sine-x	Sine-y	Remesh 5x	Remesh 10x
Base SMART	.217	.526	.379	.401
$[0, .25]$	.199	.194	.339	.371
$[0, .50]$	.190	.135	.322	.351
$[0, .75]$	.170	.099	.281	.303
$[0, 1]$	**.126**	**.077**	.265	.284
$[0, 2]$	.127	**.077**	**.251**	**.270**
$[0, 3]$	.131	.079	.256	.275

(b) Density-estimator neighborhood

k	Sine-x	Sine-y	Remesh 5x	Remesh 10x
Base SMART	.217	.526	.379	.401
4	.125	**.076**	.291	.312
8	.125	**.076**	.274	.293
16	.126	.077	.265	.284
32	.127	**.076**	**.262**	**.279**
64	**.124**	**.076**	.264	.281

Contribution of prediction consistency. Table 12 gives the condition-level values underlying Figure 4. This condition-level breakdown shows that prediction consistency lowers error for both structured redistributions and both remeshing levels in every system. The size of the gain varies across shift families, so Figure 4 reports system-level averages alongside, rather than in place of, these condition-level values.

Table 12: **Prediction-consistency ablation across physical systems.** Entries are mean $\pm$ case-level standard deviation of relative-ℓ_2 field error on unseen representation shifts. Lower is better.

Physical system	Condition	Base	DeAL without consistency	DeAL with consistency
Car aerodynamics	Sine-x	$.217 \pm .036$	$.253 \pm .015$	$\mathbf{.126 \pm .022}$
	Sine-y	$.526 \pm .028$	$.154 \pm .012$	$\mathbf{.077 \pm .007}$
	Remesh 5x	$.379 \pm .019$	$.285 \pm .010$	$\mathbf{.265 \pm .022}$
	Remesh 10x	$.401 \pm .016$	$.301 \pm .013$	$\mathbf{.284 \pm .016}$
Pump hydrodynamics	Sine-x	$.198 \pm .019$	$.172 \pm .004$	$\mathbf{.116 \pm .006}$
	Sine-y	$.169 \pm .011$	$.165 \pm .006$	$\mathbf{.107 \pm .006}$
	Remesh 5x	$.408 \pm .024$	$.245 \pm .018$	$\mathbf{.235 \pm .041}$
	Remesh 10x	$.413 \pm .022$	$.250 \pm .026$	$\mathbf{.233 \pm .022}$
Heat exchange	Sine-x	$.466 \pm .206$	$.249 \pm .044$	$\mathbf{.125 \pm .023}$
	Sine-y	$.429 \pm .032$	$.342 \pm .026$	$\mathbf{.329 \pm .014}$
	Remesh 5x	$.514 \pm .107$	$.376 \pm .074$	$\mathbf{.281 \pm .078}$
	Remesh 10x	$.461 \pm .075$	$.305 \pm .071$	$\mathbf{.256 \pm .085}$
C-core magnetostatics	Sine-x	$.341 \pm .094$	$.122 \pm .041$	$\mathbf{.097 \pm .013}$
	Sine-y	$.115 \pm .042$	$.098 \pm .005$	$\mathbf{.066 \pm .020}$
	Remesh 5x	$.142 \pm .049$	$.127 \pm .051$	$\mathbf{.117 \pm .047}$
	Remesh 10x	$.161 \pm .068$	$.148 \pm .060$	$\mathbf{.128 \pm .059}$

Loss-balancing sensitivity. Table 13 compares fixed weighting, uncertainty weighting (Kendall et al., 2018), GradNorm (Chen et al., 2018), and ConFIG (Liu et al., 2025) while holding the paired

density views, query targets, and consistency objective fixed. This isolates whether DeAL's gains depend on a particular strategy for balancing its three loss terms.

Table 13: **Loss-balancing ablation on DrivAerML with SMART.** Relative-ℓ_2 field error. All DeAL variants use the same density views and consistency objective. Base SMART and fixed weights reproduce the reference pair in Figure 2(a).

Loss balancing	Sine-x	Sine-y	Remesh 5x	Remesh 10x
Base SMART	.217	.526	.379	.401
Fixed weights	**.126**	**.077**	.265	.284
GradNorm	.128	.079	.265	.283
Uncertainty weighting	.129	.079	.266	.285
ConFIG	**.126**	.080	**.257**	**.271**

This study addresses sensitivity to optimization rather than architecture transfer. Fixed weights remain competitive for the two structured shifts, whereas ConFIG is strongest for both remeshing levels. However, the computational cost of ConFIG motivated us to stay with fixed weighting.

G Case-Level Uncertainty and Output-Domain Error

Table 14 reports mean $\pm$ case-level standard deviation and 95% intervals for the primary SMART–DeAL comparison. We use paired bootstrap intervals with 10,000 resamples. Its second panel separates surface and volume targets, showing that the aggregate result is not driven by a single output domain.

All intervals in panel (a) remain above zero, while panel (b) shows lower DeAL error for both surface and volume fields in every physical system. The panels provide complementary evidence: panel (a) quantifies uncertainty in case-level reductions, while panel (b) shows that the gains are not confined to one type of physical query.

Table 14: **SMART–DeAL uncertainty and output-domain decomposition.** (a) Condition-level means, case-level standard deviations, and 95% reduction intervals. We use paired bootstrap intervals. (b) Surface- and volume-field means across unseen shifts, reported as Base $\rightarrow$ DeAL with relative reduction in parentheses. Lower is better.

(a) Condition-level uncertainty

Task	Condition	Base	DeAL	Reduction [95% interval]
Car aerodynamics	Sine-x	$.217 \pm .036$	$\mathbf{.126 \pm .022}$	42.0% $[38.8, 45.2]$
	Sine-y	$.526 \pm .028$	$\mathbf{.077 \pm .007}$	85.4% $[85.0, 85.8]$
	Remesh 5x	$.379 \pm .019$	$\mathbf{.265 \pm .022}$	30.1% $[28.4, 31.6]$
	Remesh 10x	$.401 \pm .016$	$\mathbf{.284 \pm .016}$	29.3% $[28.1, 30.6]$
Pump hydrodynamics	Sine-x	$.198 \pm .019$	$\mathbf{.116 \pm .006}$	41.5% $[40.7, 42.2]$
	Sine-y	$.169 \pm .011$	$\mathbf{.107 \pm .006}$	36.9% $[36.3, 37.5]$
	Remesh 5x	$.408 \pm .024$	$\mathbf{.235 \pm .041}$	42.4% $[41.0, 43.8]$
	Remesh 10x	$.413 \pm .022$	$\mathbf{.233 \pm .022}$	43.5% $[42.8, 44.2]$
Heat exchange	Sine-x	$.466 \pm .206$	$\mathbf{.125 \pm .023}$	73.2% $[69.0, 76.5]$
	Sine-y	$.429 \pm .032$	$\mathbf{.329 \pm .014}$	23.3% $[21.5, 25.0]$
	Remesh 5x	$.514 \pm .107$	$\mathbf{.281 \pm .078}$	45.4% $[40.7, 50.0]$
	Remesh 10x	$.461 \pm .075$	$\mathbf{.256 \pm .085}$	44.4% $[38.8, 50.0]$
C-core magnetostatics	Sine-x	$.341 \pm .094$	$\mathbf{.097 \pm .013}$	71.7% $[69.8, 73.5]$
	Sine-y	$.115 \pm .042$	$\mathbf{.066 \pm .020}$	42.2% $[39.6, 45.0]$
	Remesh 5x	$.142 \pm .049$	$\mathbf{.117 \pm .047}$	17.4% $[14.1, 22.1]$
	Remesh 10x	$.161 \pm .068$	$\mathbf{.128 \pm .059}$	20.6% $[16.7, 24.0]$

(b) Surface and volume decomposition

Task	Surface fields	Volume fields
Car aerodynamics	$.447 \pm .019 \rightarrow .218 \pm .014$ ($\downarrow 51\%$)	$.315 \pm .018 \rightarrow .157 \pm .010$ ($\downarrow 50\%$)
Pump hydrodynamics	$.333 \pm .025 \rightarrow .188 \pm .026$ ($\downarrow 44\%$)	$.262 \pm .013 \rightarrow .158 \pm .011$ ($\downarrow 40\%$)
Heat exchange	$.583 \pm .058 \rightarrow .388 \pm .081$ ($\downarrow 33\%$)	$.352 \pm .141 \rightarrow .108 \pm .039$ ($\downarrow 69\%$)
C-core magnetostatics	$.197 \pm .060 \rightarrow .106 \pm .035$ ($\downarrow 46\%$)	$.182 \pm .055 \rightarrow .098 \pm .033$ ($\downarrow 46\%$)

H TRAINING-SEED REPLICATION

We repeat the comparison across training seeds for SMART, Geo-FNO, and PointNet++ SSG on car aerodynamics and heat exchange, and for SMART on pump hydrodynamics and C-core magnetostatics. Each Base–DeAL comparison uses seeds $\{17, 42, 73, 123, 2025\}$, producing eight task–architecture pairs, two methods, and 80 model instances. Base uses one supervised view for 300 epochs and DeAL uses two for 150 epochs, with branch losses combined before each optimizer update. We retain loss weights $(0.2, 0.2, 0.6)$, $\beta_A, \beta_B \sim \mathcal{U}[0, 1]$, and $k = 16$. Within each task–architecture pair, training cases, normalization, encoder budgets, query coordinates, and evaluation representations remain fixed. Initialization, minibatch order, and training-view sampling vary by seed; the shared seed label does not imply identical initial parameters. We evaluate the final scheduled checkpoint without selecting a checkpoint or seed by shifted-test performance.

For each run, we average Eq. (12) over the fixed test cases and then equally average Sine-x, Sine-y, mean 5x remeshing, and mean 10x remeshing. For method m, the five run-level values $E_{m,s}$ define $\overline{E}_m = \frac{1}{5}\sum_s E_{m,s}$, $\sigma_m = \sqrt{\frac{1}{4}\sum_s (E_{m,s} - \overline{E}_m)^2}$, and $\Delta = 100(1 - \overline{E}_{\mathrm{DeAL}}/\overline{E}_{\mathrm{Base}})$. All statistics are computed before rounding. Here, $\pm$ denotes sample standard deviation across training seeds, not variability across physical cases or a confidence interval.

Table 15: **Shifted-input accuracy across five training seeds.** Entries are mean $\pm$ training-seed sample standard deviation of relative-ℓ_2 field error. Reductions compare the five-seed means.

Physical system	Backbone	Base error ↓	DeAL error ↓	Reduction ↑
Car aerodynamics	SMART	$.382 \pm .004$	$\mathbf{.189 \pm .002}$	50.5%
Car aerodynamics	Geo-FNO	$1.023 \pm .010$	$\mathbf{.499 \pm .005}$	51.2%
Car aerodynamics	PointNet++ SSG	$.741 \pm .010$	$\mathbf{.417 \pm .005}$	43.7%
Heat exchange	SMART	$.469 \pm .006$	$\mathbf{.249 \pm .004}$	46.9%
Heat exchange	Geo-FNO	$.666 \pm .013$	$\mathbf{.431 \pm .014}$	35.3%
Heat exchange	PointNet++ SSG	$.628 \pm .009$	$\mathbf{.443 \pm .005}$	29.5%
Pump hydrodynamics	SMART	$.299 \pm .004$	$\mathbf{.174 \pm .003}$	41.8%
C-core magnetostatics	SMART	$.188 \pm .003$	$\mathbf{.101 \pm .002}$	46.3%

DeAL has lower aggregate error in all eight configurations and all 40 seed-matched comparisons. The five-seed reductions range from 29.5% for PointNet++ SSG on heat exchange to 51.2% for Geo-FNO on car aerodynamics. The architecture dependence is visible on heat exchange, where the reduction is 46.9% for SMART, 35.3% for Geo-FNO, and 29.5% for PointNet++ SSG. Thus, repeated training preserves the qualitative advantage without implying identical gains across backbone families.

Lower error does not imply uniformly lower training variability. For heat-exchange Geo-FNO, DeAL has a slightly larger seed standard deviation than Base (.0137 versus .0129) despite its lower mean error. Across its individual seeds, reductions range from 31.5% to 37.4%. The seed study therefore supports a persistent accuracy advantage rather than a universal reduction in sensitivity to training randomness.

Table 16: **Individual SMART scores for car aerodynamics and heat exchange.** Each value is one model's aggregate shifted-input error. The seed-42 row is the manuscript reference; the other rows are the additional-seed records underlying the five-seed summary above.

Training seed	Car: Base	Car: DeAL	Heat: Base	Heat: DeAL
17	.386	.191	.477	.252
42	.381	.188	.467	.248
73	.388	.190	.472	.254
123	.378	.187	.461	.244
2025	.379	.189	.466	.247

The individual scores show that seed effects need not preserve the same ordering between methods: car Base error is largest at seed 73, whereas car DeAL error is largest at seed 17; for heat exchange, the largest Base and DeAL errors occur at seeds 17 and 73, respectively. Nevertheless, DeAL is

lower in every comparison. Leave-one-seed-out sensitivity for the individually reported SMART pairs retains a positive reduction under every omission. These overlapping subsets are sensitivity checks, not independent replications or confidence intervals.

I GEOMETRIC FIDELITY OF REMESHING

To test representation sensitivity, remeshing must leave the underlying shape largely intact. Figure 9 visually compares native and remeshed surfaces and their local triangulations for an example from each physical system. Table 17 quantifies residual differences in surface distance, area, and normal orientation across 250 samples from each physical system.

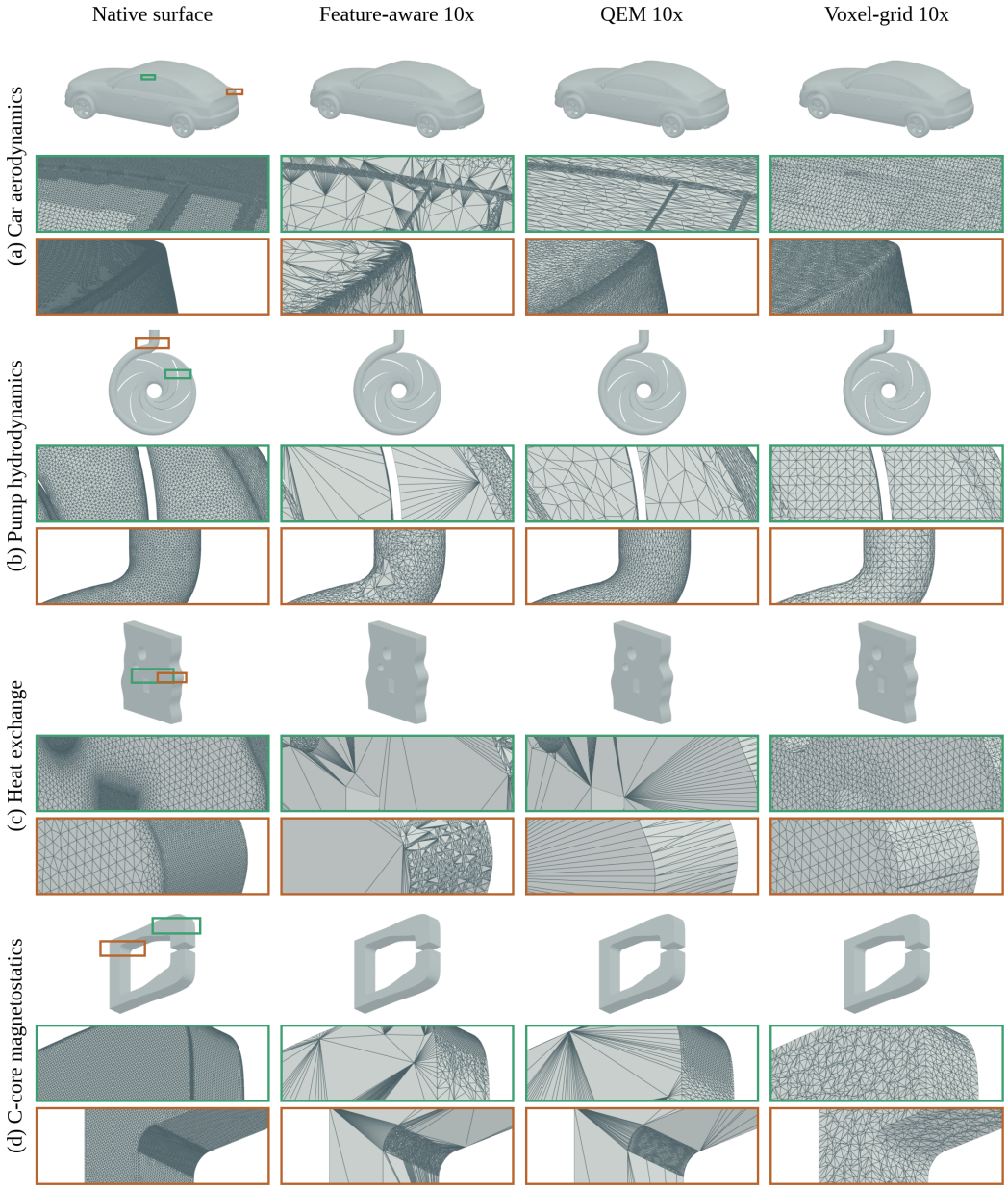


Figure 9: **Remeshing changes connectivity while leaving the overall shape recognizable.** For each system, columns show the native surface and Feature-aware, QEM, and Voxel-grid remeshes at 10x triangle-count reduction. Green and amber boxes mark two locations on the native surface. Matching borders identify their enlarged mesh views in every column, revealing changes in local triangulation.

A balanced audit covers 250 geometries per system. Each geometry is processed by Feature-aware, QEM, and Voxel-grid remeshing at both 5x and 10x reduction, yielding 1,500 geometry–remesh

records per system. Distances use 5,000 area-uniform surface samples and normal deviation uses 1,000 samples in each direction; distance metrics are normalized by the original bounding-box diagonal.

Table 17: **Geometric deviation across remeshing settings.** Entries report mean geometric deviations across 250 geometries, three remeshing methods, and two target reductions per system. Chamfer and Hausdorff$_{99}$ are percentages of the original bounding-box diagonal, area change is absolute percent, and normal deviation is in degrees. Lower is better.

Task	Chamfer	Hausdorff$_{99}$	Area change	Normal deviation
Car aerodynamics	0.0012%	0.0109%	0.288%	1.115°
Pump hydrodynamics	0.0046%	0.0429%	0.216%	1.297°
Heat exchange	0.0039%	0.0484%	0.247%	1.171°
C-core magnetostatics	0.0039%	0.0162%	0.215%	1.166°

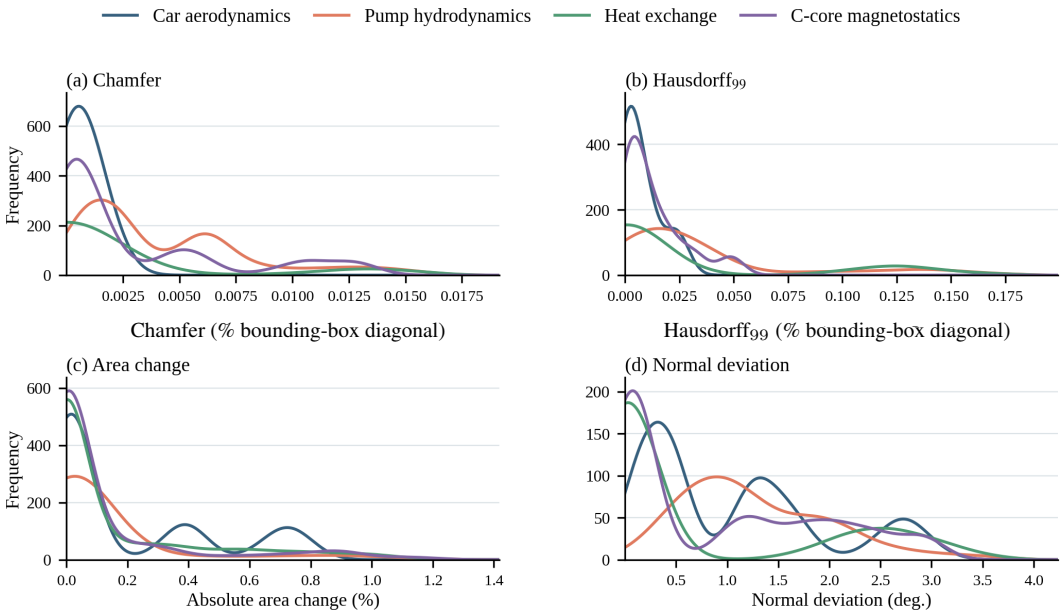


Figure 10: **Geometric deviation distributions for the remeshing audit.** Overlaid smoothed frequency curves show the 1,500 geometry–remesh records per system underlying Table 17. The vertical axes show frequency, and bin widths are shared across systems within each metric.

Across the four systems, Table 17 shows that the geometric deviations remain small relative to object scale despite the intended changes to connectivity and local sampling. Connectivity is not a preservation criterion here: changing it is the numerical intervention, whereas the physical surface remains nearly unchanged.

J Geometry, Representations, and Field Visualizations

The following figures complement aggregate errors with geometry families, surface and volume query domains, physical fields, representation changes, and spatial prediction errors.

Cross-system geometry, query domains, and fields. Figure 11 first establishes the shape families. Figure 12 then identifies their surface and volume query supports. Figure 13 shows representative physical targets on those supports, and Figure 14 summarizes their distributions. The four figures use a common row order throughout.

The atlas also makes clear why a single geometry rendering is insufficient for this benchmark. Variation is global for the cars and pumps, combinatorial for the heat-exchanger channels, and concentrated in the yoke, aperture, and gap of the C-cores. The shared row-and-column treatment exposes that diversity without suggesting that coordinates or scales are comparable across systems.

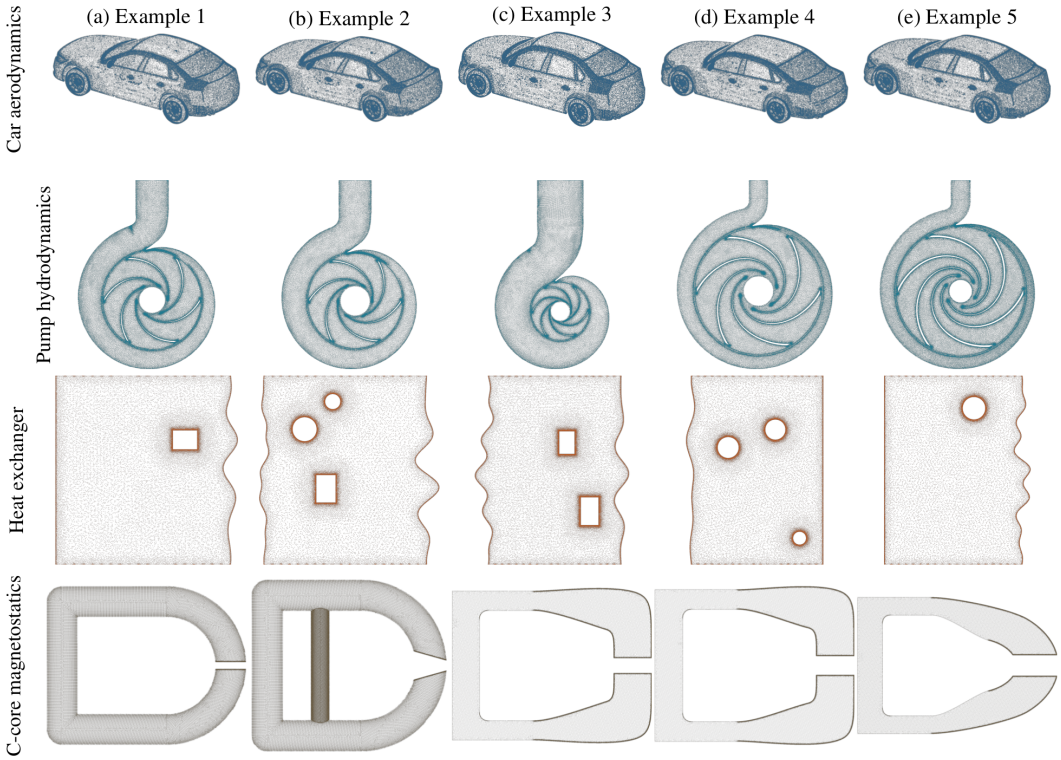


Figure 11: **Geometry diversity across the four physical systems.** Each row shows representative geometries from car aerodynamics, pump hydrodynamics, heat exchange, or C-core magnetostatics. Fixed task-specific views expose both within-task variation and the distinct shape families before representation shifts are applied.

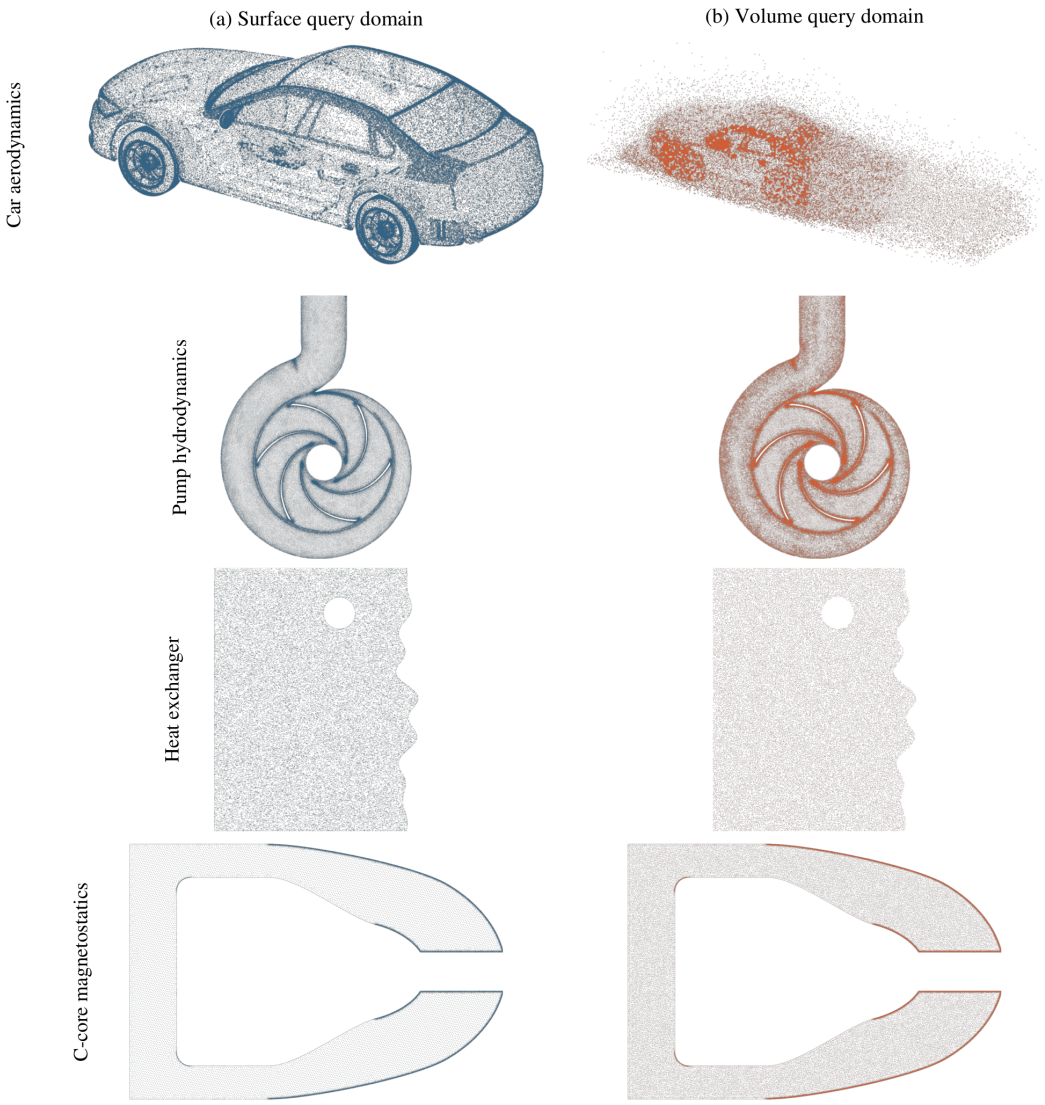


Figure 12: **Surface and volume query domains across physical systems.** (a) Surface targets lie on the physical boundary; (b) volume targets are sampled in each task's volume domain. Rows follow Figure 11. The C-core model predicts within the solid core, while the surrounding air region is used only by the numerical solver. Surface and volume errors are computed on their own domains before being combined with equal weight.

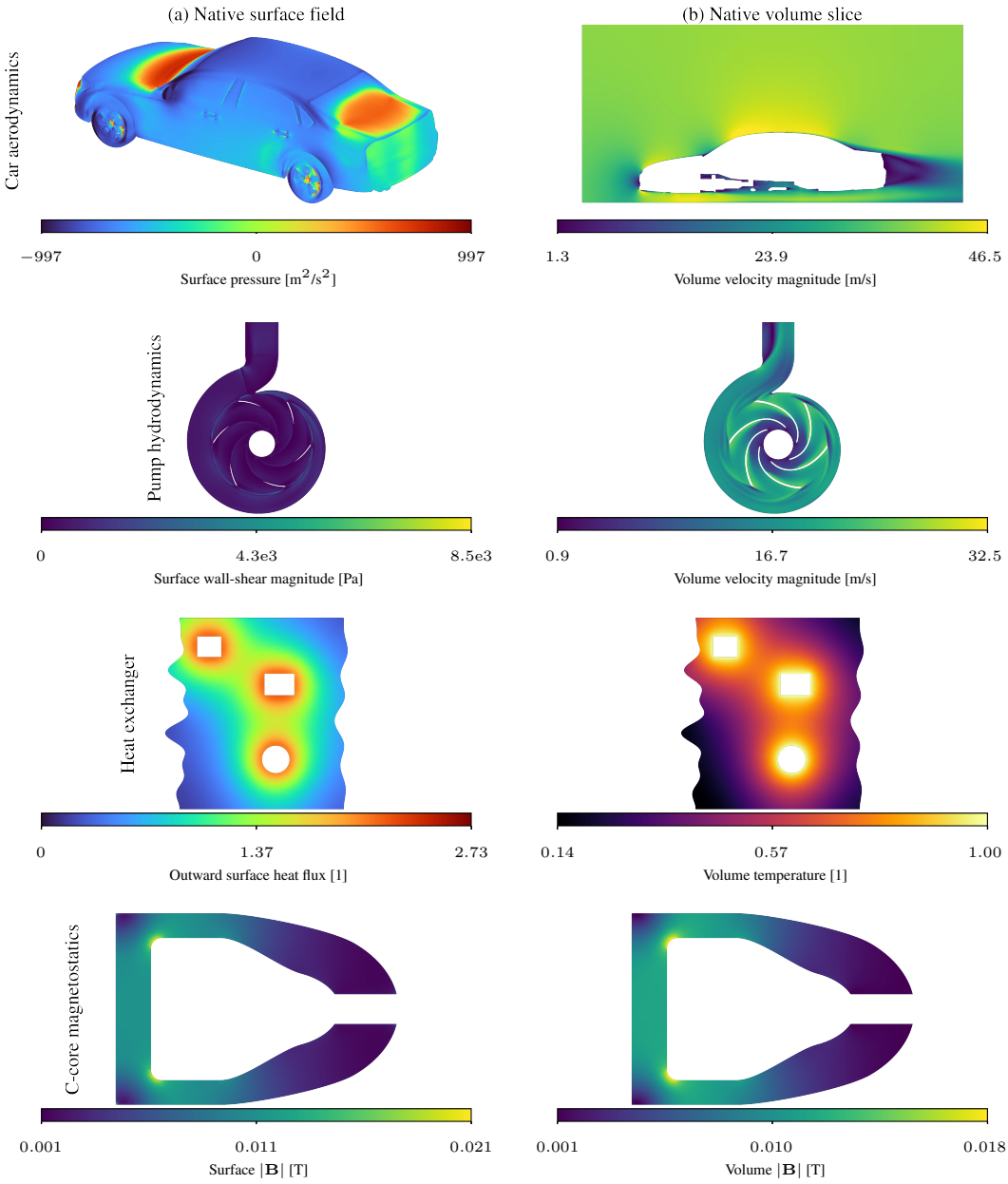


Figure 13: **Representative physical targets in both output domains.** (a) Boundary fields on the native surface mesh; (b) planar cuts through the native CFD or FEM volume meshes. Each row uses the same physical case and coordinate frame in both columns. The pump plane crosses the impeller mid-span, revealing the blade passages, volute, and outlet. For heat exchange, the surface panel shows the exterior Robin flux, while the three channel walls remain as holes. Each colorbar uses the robust 1st–99th-percentile range of its panel's finite values, symmetrized about zero for signed fields; values outside the displayed range are clipped.

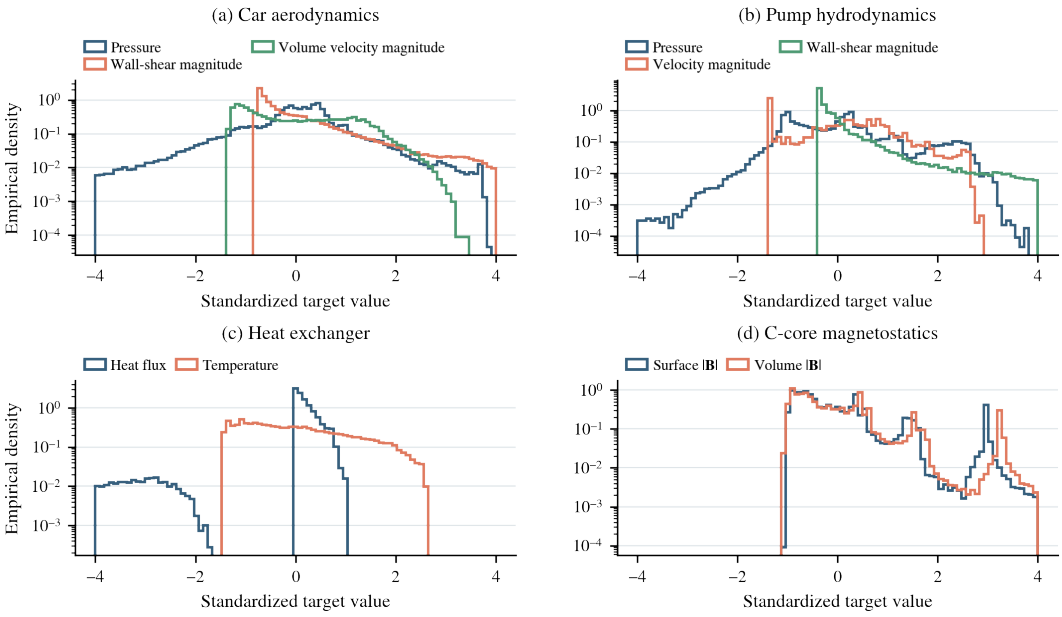


Figure 14: **Target-field distributions across four systems.** Curves pool representative cases. Each pressure curve combines equal surface and volume samples for car and pump. Log densities reveal skew and tails in standardized fields without comparing physical units.

Figures 15–18 relate task-specific geometry to physical responses across all four systems. The C-core aperture and pole constructions are analyzed separately.

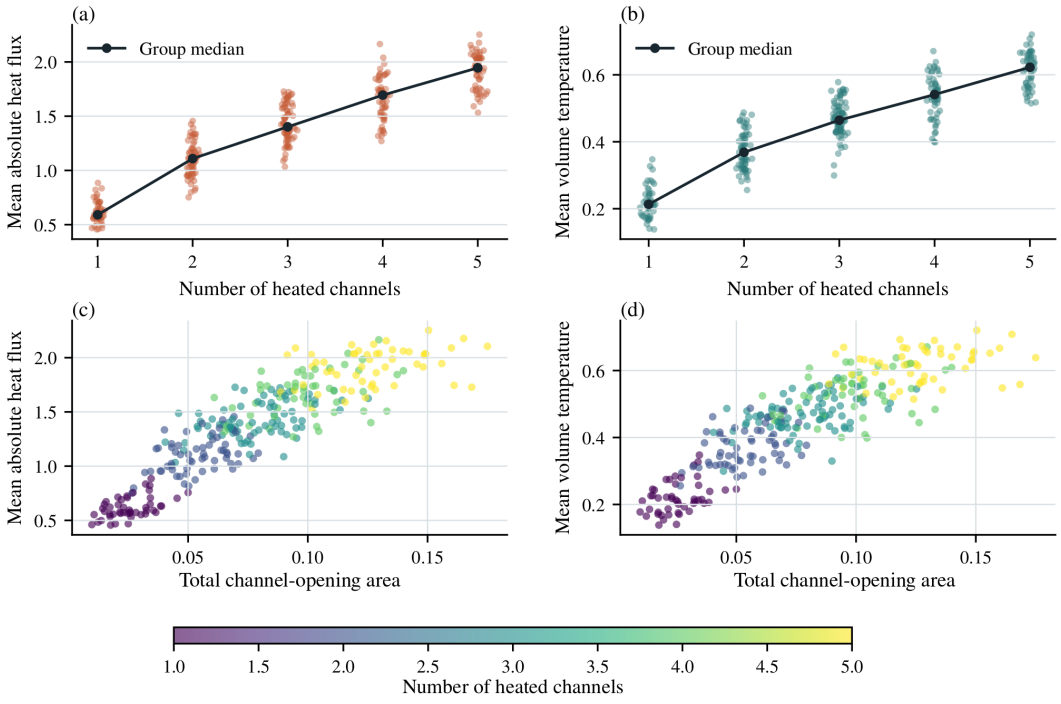


Figure 15: **Deterministic geometry–physics coupling in the heat-exchanger benchmark.** Each point is one generated case and color gives its heated-channel count. Channel count and total channel-opening cross-sectional area track the finite-element heat flux and temperature, confirming that the targets vary with observable geometry under fixed physics.

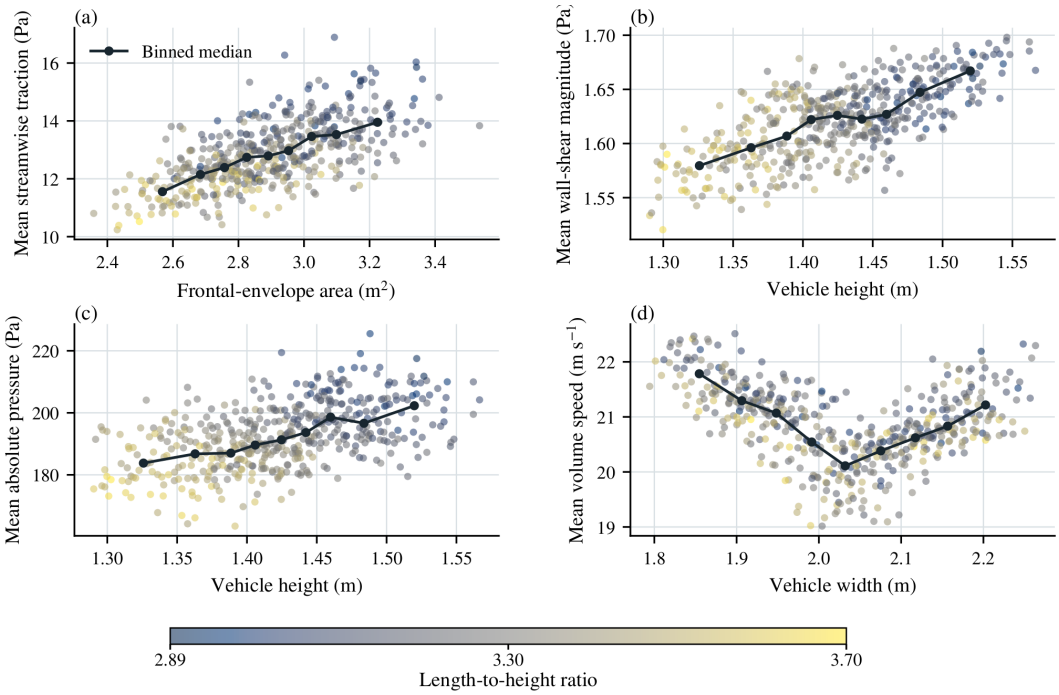


Figure 16: **Geometry–physics coupling in DrivAerML.** Frontal-envelope area, height, and width are computed from the native surface coordinates. Surface responses are quadrature-weighted means. Streamwise traction combines pressure and wall shear. The volume response averages speed at the released physical queries. Color gives the vehicle length-to-height ratio, and the dark curve is an equal-count binned median. The shared flow setup means that the variation shown here is geometric.

The median trends show that larger frontal envelopes carry larger mean streamwise traction, while vehicle height organizes both pressure and wall shear. The width–speed relation is non-monotone, which is also informative: no single global dimension explains the complete flow field. These panels are dataset diagnostics rather than surrogate-performance evidence.

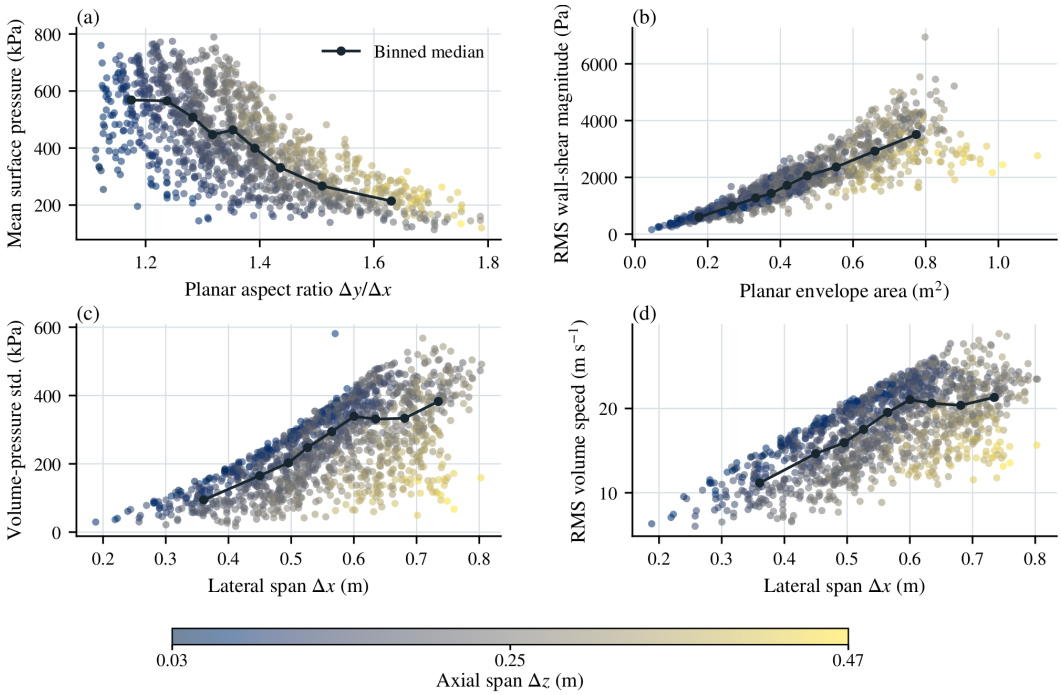


Figure 17: **Geometry–physics coupling in SHIFT-Pump.** Every horizontal-axis descriptor is computed from the realized surface coordinates. The responses summarize the released CFD fields: (a) mean surface pressure, (b) root-mean-square (RMS) wall-shear magnitude, (c) volume-pressure standard deviation, and (d) RMS volume speed. Color gives the axial span, and the dark curve is an equal-count binned median.

The pump cases also occupy structured geometry–field regimes. Planar aspect ratio organizes mean pressure, planar envelope area strongly tracks wall shear, and lateral span tracks both volume-pressure variation and flow speed. As in Figure 16, these associations characterize the dataset rather than surrogate accuracy.

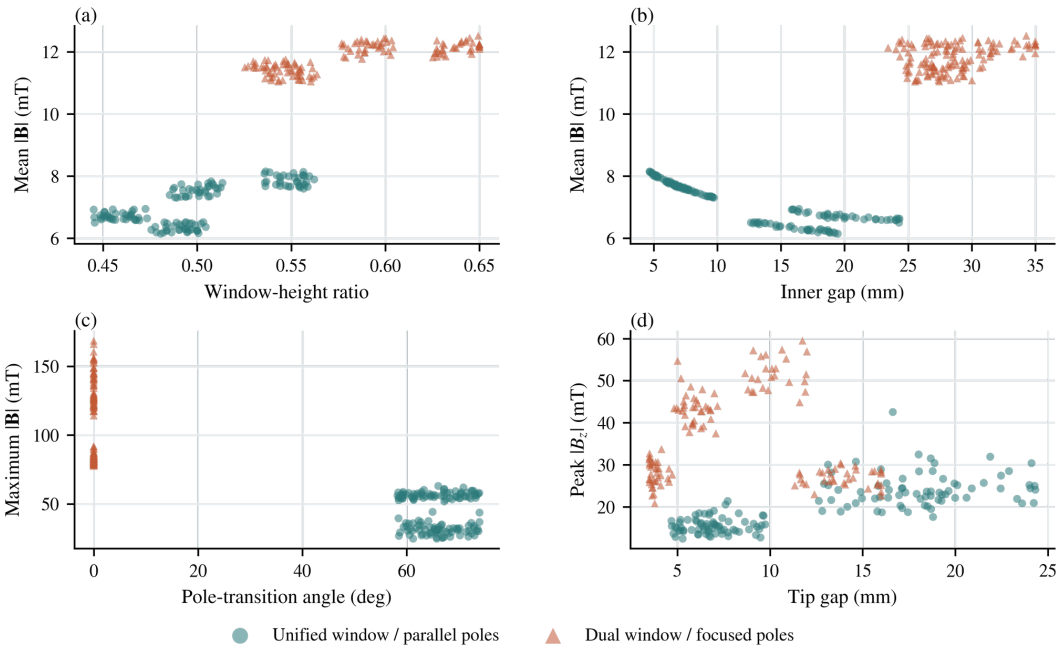


Figure 18: **Geometry–physics coupling in C-core magnetostatics.** Native finite-element summaries relate aperture ratio, inner gap, pole-transition angle, and tip gap to mean, maximum, or out-of-plane magnetic flux density. Color and marker distinguish the unified-window/parallel-pole and dual-window/focused-pole constructions. Raw cases are shown without fitted trends because the topology-specific geometry families need not vary continuously with one another. Excitation and material properties are fixed, so the displayed field variation follows the generated core geometry.

The two construction families occupy distinct magnetic-response regimes. Within each family, aperture and gap variables organize the mean and peak fields, while pooling the families would create an artificial interpolation between topologically different cores. We therefore do not fit a common curve across the two topologies.

Controlled representation variations. We visualize the continuous inverse-density training path and axis-structured evaluation views in Figure 19, while Figure 20 contrasts it with the matched downsampling and local-masking controls.

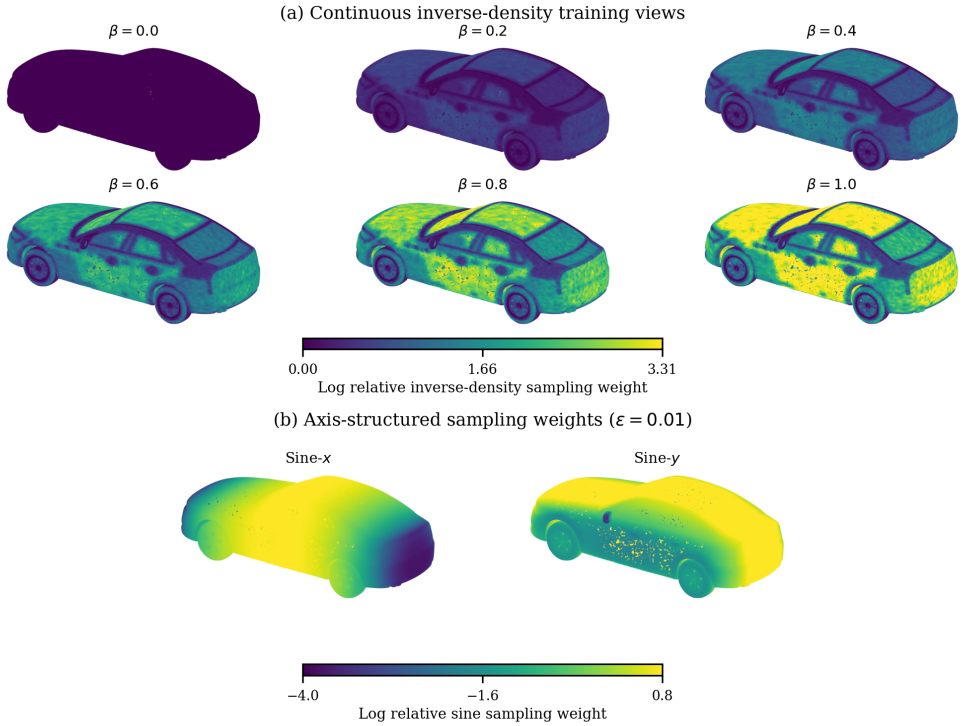


Figure 19: **Continuous inverse-density training views and axis-structured evaluation views.** (a) The encoder budget and underlying geometry remain fixed as β increases from 0 to 1. Larger β progressively reallocates samples from dense regions toward sparsely represented regions. Color maps the log relative inverse-density sampling weight. (b) Sine-x and Sine-y use sampling weights $\sin^2(\pi z) + 10^{-2}$ at the same point budget. Color maps their log relative sampling weight.

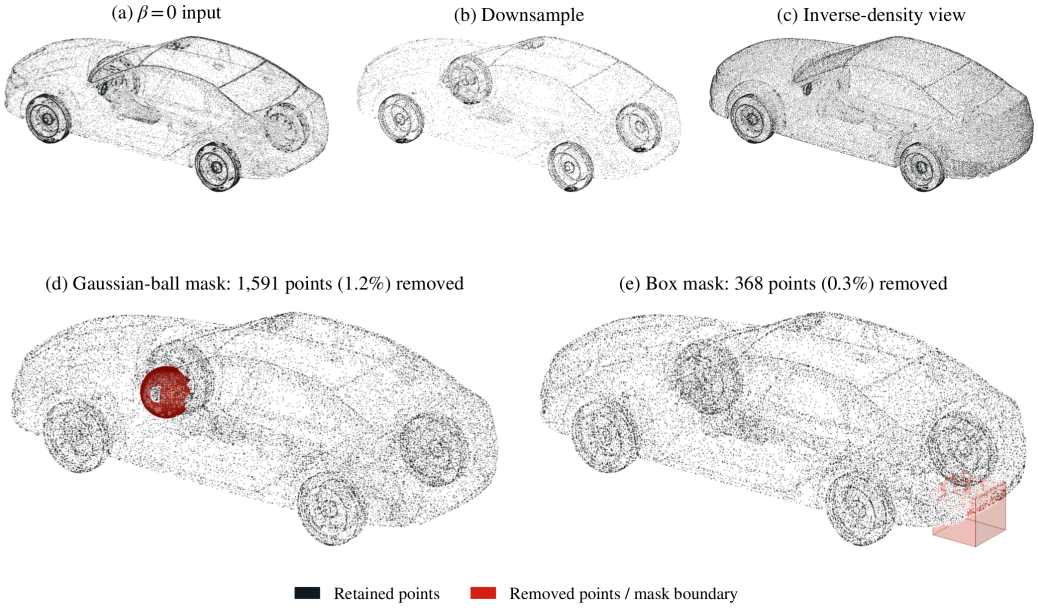
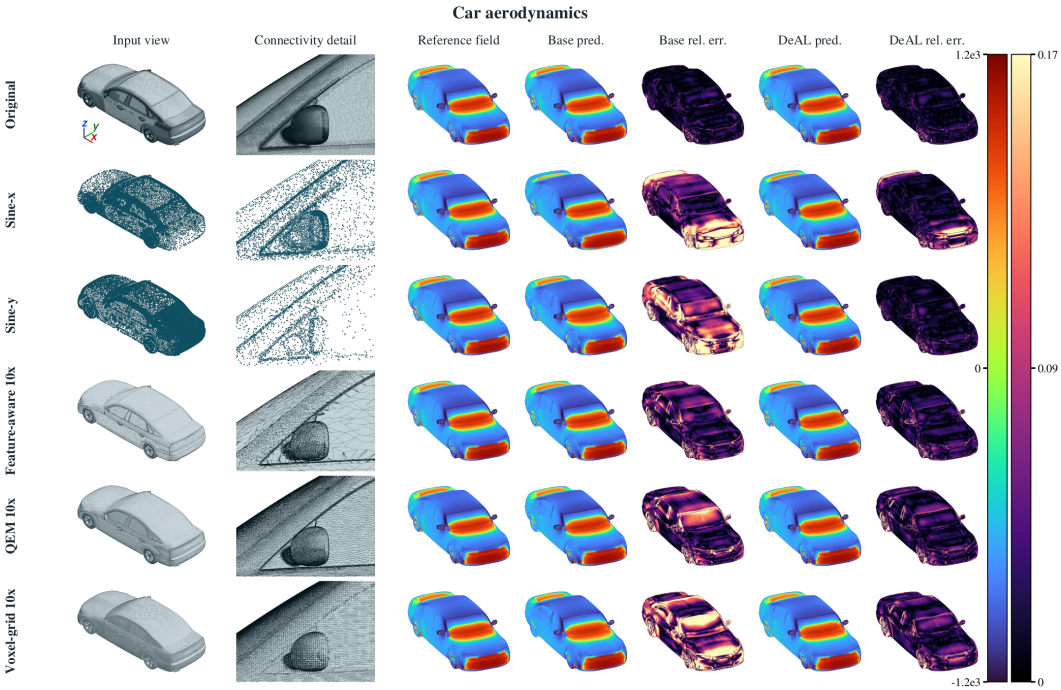


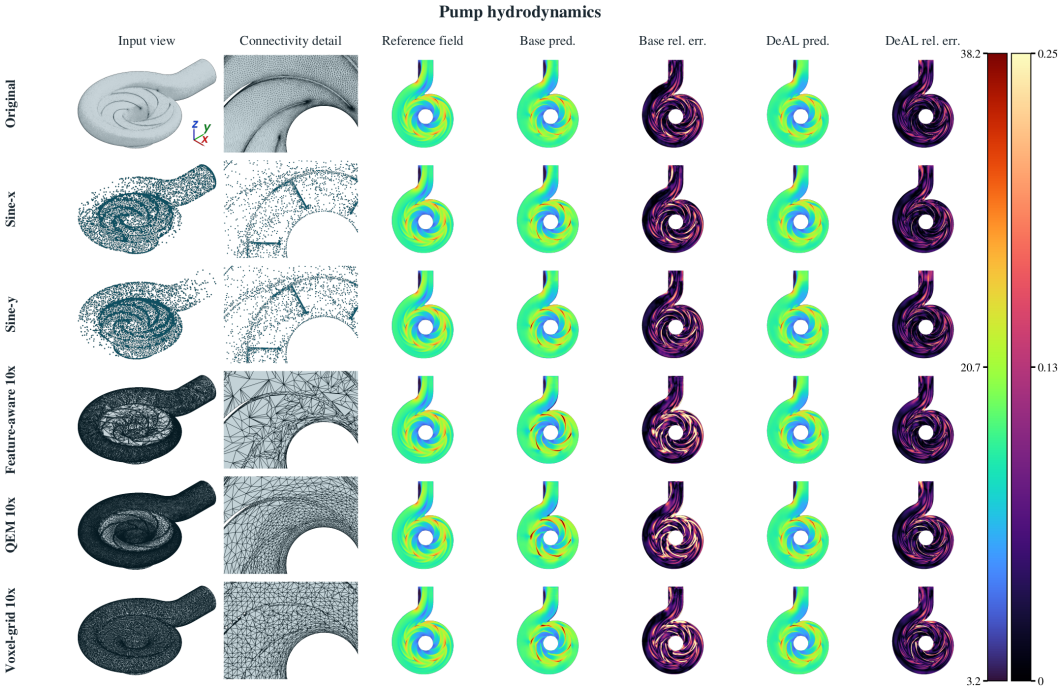
Figure 20: **Training-view generators compared on the same surface.** Downsampling reduces the point budget globally; Gaussian-ball and box masking remove localized regions, with removed samples shown in red and the mask support outlined; inverse-density sampling retains global surface support at a fixed point budget without explicitly masking a region while continuously changing its empirical density. These are the view generators used by the controlled strategy comparison.

Spatial effects of individual representation shifts. Figures 21–24 resolve the individual input representations behind the qualitative examples. Within each row, the physical case, target field, and native query locations remain fixed. Only the encoder representation changes. Sine rows display their sampled point clouds, while remesh rows display the connected 10x input surfaces.



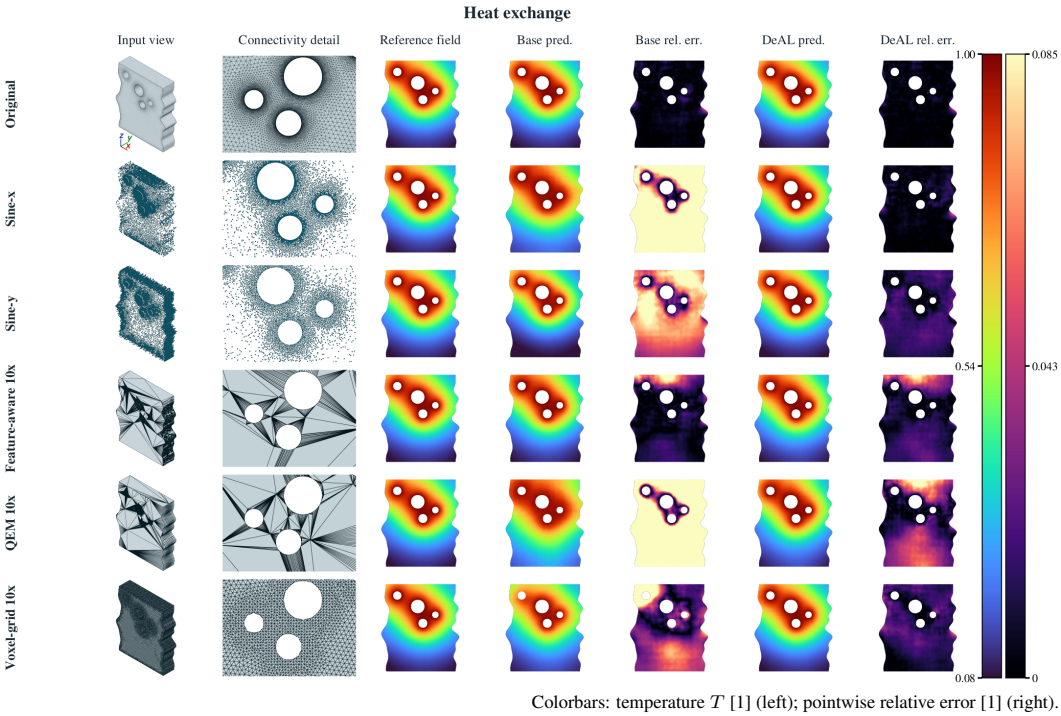
Colorbars: surface pressure p [m^2/s^2] (left); pointwise relative error [1] (right).

Figure 21: **Condition-level spatial predictions for car aerodynamics.** Each row evaluates one held-out encoder representation of the same car. The left column shows the sine point-cloud input or the actual 10x remesh surface. Reference, Base, and DeAL fields are decoded at the same native surface queries and share one field scale. The two relative-error columns share one linear scale. Across the shown shifts, DeAL reduces the prominent local error regions relative to Base.



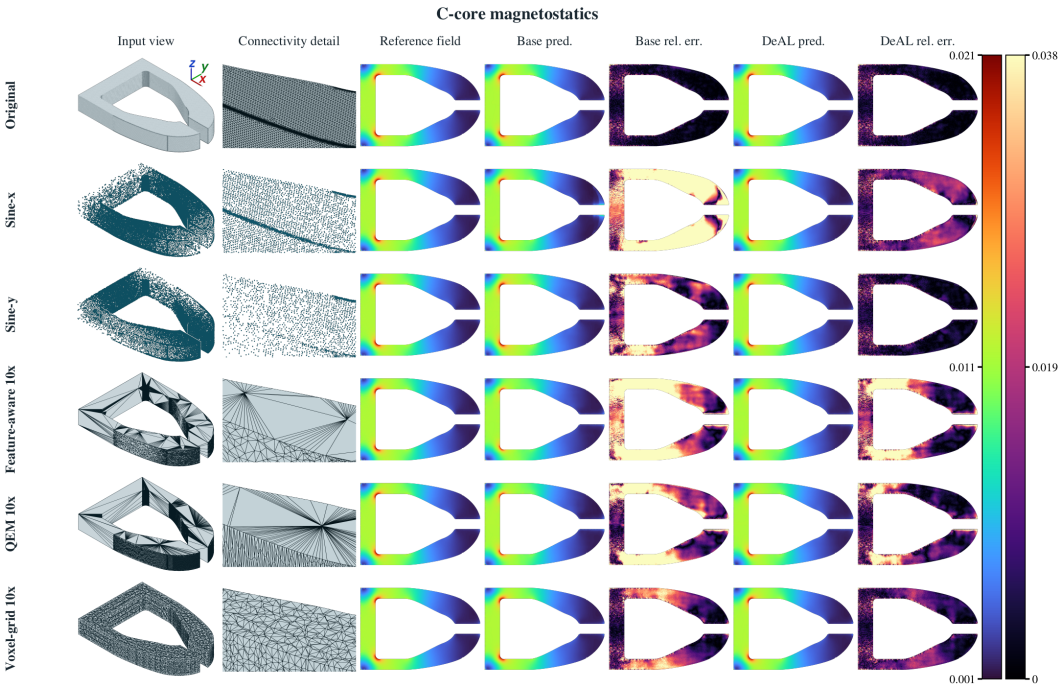
Colorbars: velocity magnitude $|\mathbf{u}|$ [m/s] (left); pointwise relative error [1] (right).

Figure 22: **Condition-level spatial predictions for pump hydrodynamics.** Each row evaluates one held-out encoder representation of the same pump. The left column shows the sine point-cloud input or the actual 10x remesh surface. Reference, Base, and DeAL fields are decoded on the same native blade-midspan slice and share one field scale. The two relative-error columns share one linear scale. Across the shown shifts, DeAL reduces the prominent local error regions relative to Base.



Colorbars: temperature T [1] (left); pointwise relative error [1] (right).

Figure 23: **Condition-level spatial predictions for heat exchange.** Each row evaluates one held-out encoder representation of the same heat exchanger. The left column shows the sine point-cloud input or the available 10x remesh surface. Reference, Base, and DeAL fields are decoded on the same native central slice and share one field scale. The two relative-error columns share one linear scale. Across the shown shifts, DeAL reduces the prominent local error regions relative to Base.



Colorbars: flux-density magnitude $|\mathbf{B}|$ [T] (left); pointwise relative error [1] (right).

Figure 24: **Condition-level spatial predictions for C-core magnetostatics.** Each row evaluates one held-out encoder representation of the same C-core geometry. The left column shows the sine point-cloud input or the actual 10x remesh surface. Reference, Base, and DeAL fields are decoded at the same native surface queries and share one field scale. The two relative-error columns share one linear scale. Across the shown shifts, DeAL reduces the prominent local error regions relative to Base.

Figure 25 then resolves the aerodynamic volume more finely. DeAL and Base are compared on the same native cells and physical query vertices, so apparent spatial differences do not come from displaying different point samples.

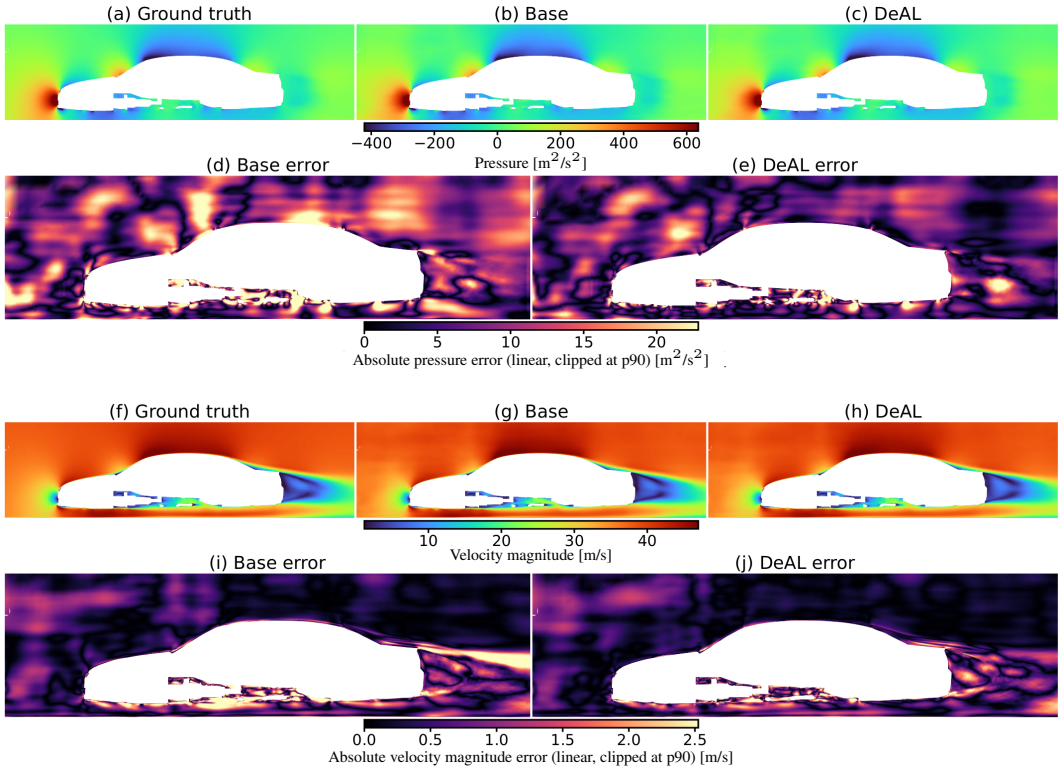


Figure 25: **Native-mesh volumetric DrivAerML inference under a QEM 10x encoder input.** Base SMART and DeAL are decoded at all 169,257 vertices of the official $y = 0$ CFD slice. Its 164,452 source polygons provide the ground-truth pressure and velocity and are triangulated only for rendering. For each field, ground truth, Base, and DeAL share one physical scale, while the two absolute error maps share a linear scale from zero to their pooled 90th percentile. The white silhouette is the solid vehicle region excluded from the native fluid mesh. Pressure error is concentrated around the stagnation region, underbody, roof transition, and wake. Velocity-magnitude error is strongest in the underbody and separated wake. Across both fields, the side-by-side error maps show that DeAL reduces these errors relative to Base while using the same remeshed encoder input and native query locations.