DROP-OP: GENERALIZED PASSIVE RESPONSE FOR OBJECT DYNAMICS PREDICTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Predicting object dynamics from sparse interactions requires a representation that captures how responses vary across space and actions. We introduce generalized passive response (GPR), an action-to-displacement response map defined for a given initial state and interaction interface. DROP-Op identifies a finite-dimensional GPR state by fitting spatially varying coefficients in a learned response basis. The fit combines geometry-based evidence sharing with response-dependent regularization, and the identified state is reused to predict responses to new actions under compatible conditions. We evaluate DROP-Op on a simulated grasping benchmark with individual and combined shifts in geometry, material properties, actions, and grippers. On the 46-panel confirmation set, DROP-Op reduces mean endpoint and translation-centered displacement errors by 31.6% and 30.9%, respectively, relative to ALPaCA-MSE, the baseline with the lowest mean endpoint error. Matched attention and pooled-identification controls also yield higher prediction errors. Additional backbone and external-task studies show task-dependent benefits of adapting the identification procedure. Reusing the identified state amortizes identification cost over repeated queries.

1 INTRODUCTION

Predicting an object's motion and deformation under new interactions requires accounting for differences in geometry, material properties, and interaction conditions. Existing models encode physical information through geometry and material descriptors or learn it from observed interactions (Zhang et al., 2024; Xu et al., 2019). However, selecting task-relevant physical properties and estimating their spatial distributions can require substantial effort (Liang et al., 2020; Memmel et al., 2024; Dagli et al., 2026). This motivates representations that capture the combined effects of these properties directly from an object's observed behavior.

Learning from interactions offers an alternative to explicitly estimating physical properties, but its effectiveness depends on how the observed responses are represented and used. Context models may discard useful response details during encoding or fail to exploit the retained information during prediction (Kim et al., 2019). Reprocessing the same interactions for every new query also introduces redundant computation. The challenge is therefore to organize sparse interaction observations into a representation that preserves spatial response information and can be reused for prediction under new actions.

In this work, we introduce generalized passive response (GPR), which describes an object's responses to external actions under a specified initial state and interaction interface. GPR maps actions to spatiotemporal displacement fields, capturing the combined effects of geometry, materials, and interaction conditions without requiring separate estimates of each relevant physical property. For prediction, we estimate a finite-dimensional *GPR state* from sparse samples of this response map.

We develop DROP-Op to identify this state and use it for new-action prediction. DROP-Op fits spatially varying coefficients in a learned response basis, shares response evidence across geometrically related locations, and stabilizes the fit through response-dependent regularization. The fitted coefficients and geometry features form the identified GPR state. For each new action, the model combines action-dependent response features with the fitted coefficients to predict the displacement field. The same state can be reused without repeating identification, provided that the initial state and interaction interface remain compatible.

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.

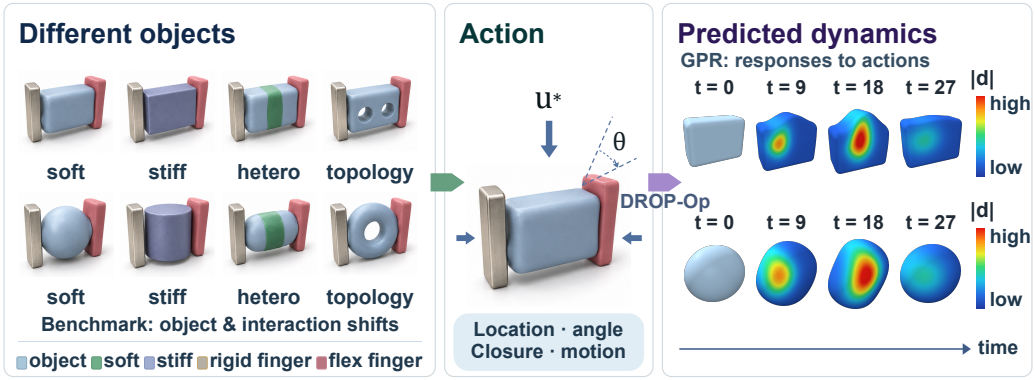

Figure 1: **Predicting object dynamics under interaction.** Object geometry, material layout, topology, and gripper configuration affect the response to an action u^*. The schematic sequences illustrate spatially varying displacement magnitude $|d|$ over time. Our benchmark evaluates prediction across individual and combined object and interaction shifts, while DROP-Op uses sparse observed interactions to identify a GPR state for subsequent prediction.

Evaluating generalization across objects and interactions is complicated by differences in the prediction tasks and evaluation settings used in existing studies (Huang et al., 2023; Shen et al., 2022). We therefore introduce a simulated grasping benchmark that provides a common task and observation budget for comparing models (Figure 1). Given an object's initial geometry and a few observed grasps, a model predicts its motion and deformation under a new grasp. The benchmark varies geometry, materials, actions, and gripper configurations individually and jointly, enabling controlled evaluation of out-of-distribution prediction.

On the confirmation set, DROP-Op achieves the lowest mean displacement errors and highest mean surface-overlap scores among the evaluated parameter-matched baselines. Controlled representation comparisons favor the identified GPR state within the evaluated prediction frameworks, while matched pooling and direct-attention controls support the benefits of spatial response identification. A separate study also shows lower errors than the evaluated expanded baselines at comparable FLOP budgets under a fixed training protocol. Further experiments demonstrate computational savings from state reuse and improvements on the tested backbones, while external-task adaptations show both gains and negative transfer.

Our contributions are:

1. **A controlled grasping benchmark** for evaluating dynamics prediction under individual and combined object and interaction shifts, with a common observation budget.
2. **GPR**, a response-based object representation that describes spatiotemporal behavior under specified initial states and interaction interfaces, reducing reliance on separately estimated physical properties.
3. **DROP-Op**, a prediction framework that identifies finite-dimensional GPR states through spatial response fitting, geometry-based evidence sharing, and response-dependent regularization, and reuses these states across compatible queries.

2 Related Work

Physical object representations. Physical representations encode material properties explicitly or learn physical information from behavior. NeRF2Physics estimates physical properties from visual and language features (Zhai et al., 2024), while VoMP predicts volumetric fields of Young's modulus, Poisson's ratio, and density (Dagli et al., 2026). DensePhysNet learns dense spatial features through repeated interactions (Xu et al., 2019), while latent dynamics models and neural force fields support prediction without separately estimating every material property (Daniel et al., 2026; Li et al., 2026b). GPR captures the combined effects through spatiotemporal responses. We compare its identified state with explicit-property inputs and learned representations.

Context-based dynamics and system identification. Observed interactions support dynamics adaptation through parameter identification or context conditioning. Task-oriented system identification selects informative interactions to estimate parameters relevant to downstream control (Liang et al., 2020; Memmel et al., 2024). IIDA infers latent physical information from context, whereas AdaptiGraph conditions predictions on identified material properties (Evans et al., 2022; Zhang et al., 2024). CNP pools observations (Garnelo et al., 2018), ANP uses attention-based conditioning (Kim et al., 2019), and ICON learns function mappings from examples (Yang et al., 2023). Alternatively, coefficients can be fitted in learned feature spaces through Bayesian regression or differentiable ridge adaptation (Harrison et al., 2018; Bertinetto et al., 2019). DROP-Op organizes observations into spatial response equations with geometry-based sharing and response-dependent regularization.

Neural operators and model construction. Neural operators learn mappings between input and output functions (Lu et al., 2021; Li et al., 2021; Kovachki et al., 2023). GNOT and Transolver accommodate heterogeneous inputs and spatial structure through different aggregation mechanisms (Hao et al., 2023; Wu et al., 2024). Mesh- and contact-based models incorporate spatial relations into deformable dynamics prediction (Pfaff et al., 2021; Saleh et al., 2024; Feng et al., 2026), while DefGraspNets predicts stress and deformation fields for grasp planning (Huang et al., 2023). Complementary work assists model construction: Code as Policies generates embodied-control programs, and CodePDE generates numerical solvers for specified PDE problems (Liang et al., 2023; Li et al., 2026a). These methods assist implementation and refinement from task specifications. DROP-Op instead focuses on organizing interaction evidence into an identified response state, which a learned predictor reuses across compatible new actions.

Physical generalization benchmarks. Physical benchmarks evaluate object understanding, event prediction, and interactive problem solving (Gao et al., 2023; Bear et al., 2021; Li et al., 2024), while PDEBench compares field predictors on differential-equation problems (Takamoto et al., 2022). Our grasping benchmark evaluates whether sparse object responses enable new-action prediction by testing object, action, and interface shifts individually and jointly under a fixed observation budget.

3 METHOD

Given a few observed interactions, DROP-Op identifies a spatial GPR state and uses it to predict the object's response to new actions (Figure 2).

3.1 GPR: REPRESENTING AN OBJECT THROUGH ITS RESPONSES

An object's response to interaction reflects the combined effects of its geometry, materials, internal structure and interaction conditions. For a given initial state and interaction setting, we define generalized passive response (GPR) as the mapping from admissible external actions to the object's displacement fields:

$$\mathcal{G}_o : u \longmapsto d_o(\cdot, \cdot; u). \tag{1}$$

Here, $d_o(x, t; u)$ is displacement at position x and time t under action u. Each interaction samples this spatiotemporal response map, from which DROP-Op identifies a compact state for new-action prediction. "Passive" refers to response to external excitation (Appendix A.2).

3.2 DROP-OP: IDENTIFY ONCE, PREDICT NEW ACTIONS

Start with shared response patterns. The network learns a shared displacement prediction μ_θ and a response basis ϕ_θ. The basis provides adjustable patterns of motion over space and time. Object-specific coefficients $a_o(x)$ determine their contribution:

$$\widehat{d}_o(x, t; u) = \mu_\theta(x, t; g_o, u) + \phi_\theta(x, t; g_o, u)^\top a_o(x). \tag{2}$$

The geometry g_o and action u determine the patterns. The observations determine the coefficients, allowing the same learned predictor to adapt to different objects.

Fit the current object's responses. We fit coefficients to the difference between observed responses and the shared prediction. Each anchor combines its own observations with those from geometrically

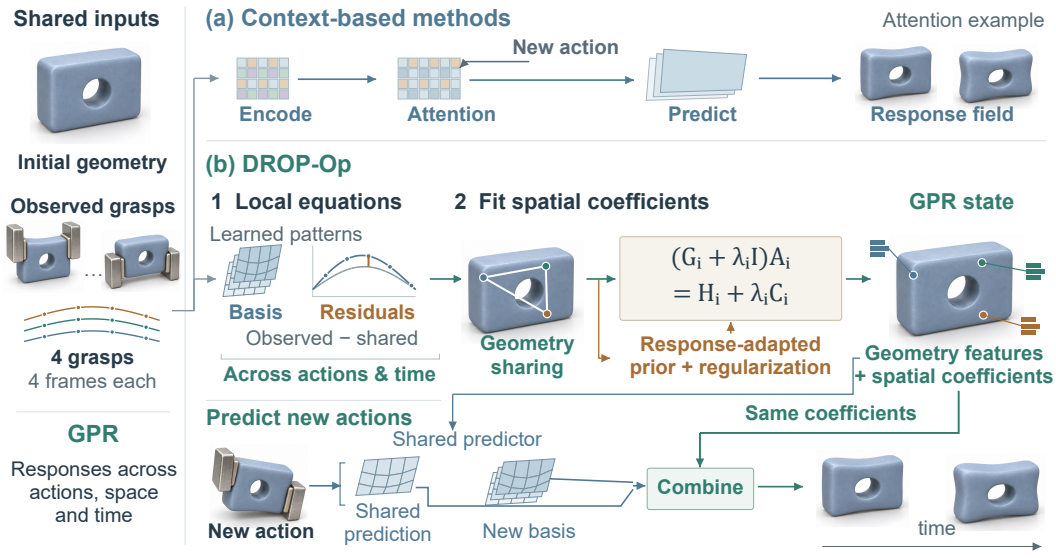

Figure 2: **Context-based prediction and DROP-Op.** (a) Attention-based conditioning. (b) DROP-Op organizes observed responses, fits spatial coefficients with geometry-based sharing and response-dependent regularization, and reuses the GPR state for new actions (Appendix A.3).

related anchors. Let Q_i contain the weighted basis features and Y_i the corresponding response residuals:

$$A_i = \arg\min_A \|Q_i A - Y_i\|_F^2 + \lambda_i \|A - C_i\|_F^2. \tag{3}$$

The first term fits the observations. The second stabilizes the coefficients around a learned reference C_i. Both C_i and the positive strength λ_i depend on geometry and observed responses. Setting the gradient to zero gives

$$(G_i + \lambda_i I)A_i = H_i + \lambda_i C_i, \qquad G_i = Q_i^\top Q_i, \quad H_i = Q_i^\top Y_i. \tag{4}$$

Solving this system at each anchor gives spatial coefficients. These coefficients and the geometry features form the identified GPR state (Appendix A.3).

Predict and reuse. For a new action, Eq. 2 uses coefficients interpolated from anchors to prediction points. Training minimizes prediction error through the coefficient fit. At inference, network weights stay fixed, and compatible queries reuse the identified state. Centroid motion and deformation are computed from the predicted field. Appendix A.5 gives the full cost, and Appendix G.2 defines the readouts.

3.3 Advances over existing representations and operators

Context models learn from interactions (Garnelo et al., 2018; Yang et al., 2023), while learned-feature regression supports rapid adaptation (Harrison et al., 2018; Bertinetto et al., 2019). DROP-Op combines spatial fitting, geometry-based evidence sharing and response-dependent regularization in a reusable object state. Sections 4.3 and 4.4 test this design against fixed and learned response bases, attention conditioning and pooled identification.

4 Experiments

We evaluate DROP-Op's accuracy and computational cost, then compare GPR with alternative representations. We examine how response identification improves on context-based prediction, whether the method benefits other backbones and tasks, and how an identified state can be reused.

4.1 Grasping benchmark and evaluation setup

Prediction task and observations. Models predict an object's motion and deformation under a new grasp from its initial shape and four observed interactions (K4). Each interaction includes grip-

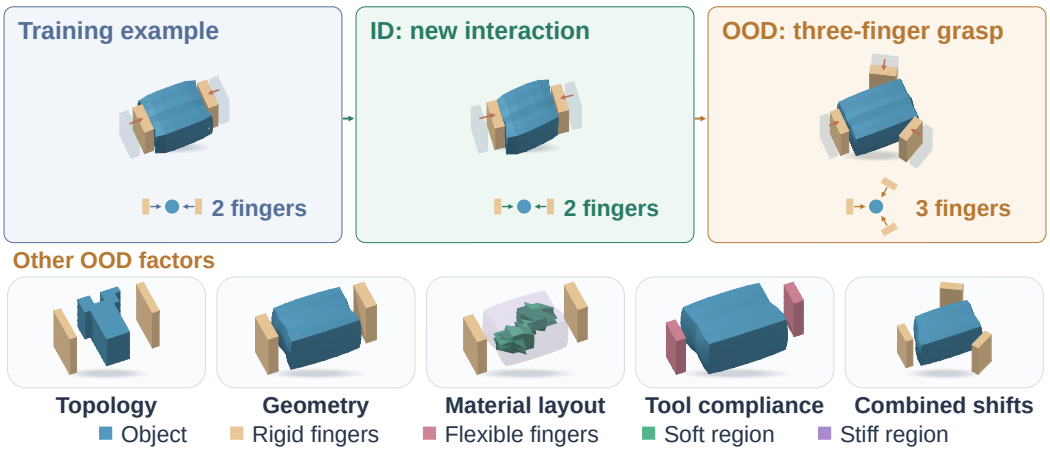

Figure 3: **In-distribution (ID) and out-of-distribution (OOD) interactions.** Top: two-finger training and ID grasps versus a three-finger OOD grasp. Translucent fingers and arrows show initial poses and motion. Bottom: individual and combined shifts (Appendix B.3).

per motion and responses at 32 points over four frames. MuJoCo simulations (Todorov et al., 2012) provide training targets. Predictions cover 96 points and 28 frames, with frames 1–27 evaluated.

Generalization factors and difficulty. Out-of-distribution (OOD) tests vary *objects* (shape, size, topology, stiffness and material layout), *actions* (location, angle, closure and timing), and *grippers* (finger count, geometry and compliance), individually and jointly (Figure 3). Four levels organize the difficulty. *V1-A* starts with geometry and actions only (K0). *V1-B* adds K4 evidence and tests instance, mechanics, shape and topology shifts with familiar and new actions. *V2-L1* retains this hierarchy with more complex shapes and material layouts. *V2-L2* adds contact, motion-rate, actuation, gripper and spatial-mechanics shifts, including combined changes. Appendix B.1 details the hierarchy (Table 6).

Training and evaluation splits. V1 and V2 models train separately. V2 training contains 448 identities, 128 meshes and 14 geometry families, with four support and four query trajectories per identity. The development set covers 48 evaluation conditions, called panels. The confirmation set is evaluated after model freezing. V2 confirmation covers 46 panels (30 L1, 16 L2), of which 44 test unseen identities and two test new queries for seen identities. Appendices B.1–B.2 detail generation, simulation and splits.

Training protocol. Main-table predictors use K4 and train for 138 epochs with three seeds. The shared-predictor representation comparison and expanded-baseline study use one seed. Backbone transfer uses three adapter initializations per fixed backbone. Experiment-specific sample counts and protocols appear in Appendices C.3–G.

Metrics. Endpoint error (EPE) measures displacement error in mm, and dEPE removes mean translation. Intersection over union (IoU, %) compares predicted and ground-truth surface regions. Deformation IoU (dIoU, %) applies the same ratio to regions changed relative to the initial surface. We evaluate surface-tube halfwidths of 2, 4 and 8 mm and report 2 mm in the main text (Appendix B.2.2). Unless stated otherwise, we average over queries within each identity, then over identities within each panel, giving all panels equal weight. MSE and NLL denote mean squared error and negative log-likelihood training objectives.

4.2 Generalization and computational efficiency of DROP-Op

On the confirmation set, DROP-Op achieves the best average trajectory and shape prediction accuracy across changes in objects, actions and interaction conditions (Table 1). It reduces EPE and dEPE by 31.58% and 30.86% relative to ALPaCA-MSE, the strongest baseline by mean EPE. Figure 4 vi-

Table 1: **Generalization across object and interaction shifts.** Bold marks the best prediction scores. Cold cost includes identification and prediction. Parameters are in thousands.

Method	Error (mm) ↓		Overlap (%) ↑		Cost	
	EPE	dEPE	IoU	dIoU	Params. (k)	Cold GFLOPs
ALPaCA-MSE (Harrison et al., 2018)	1.134	1.044	80.0	64.9	333.1	2.111
ICON (Yang et al., 2023)	1.196	1.060	77.7	60.2	338.0	20.093
DeformContact (global) (Saleh et al., 2024)	1.282	1.183	77.1	60.7	341.7	7.212
DeformContact (local) (Saleh et al., 2024)	1.327	1.218	76.3	59.8	341.7	7.226
ALPaCA-NLL (Harrison et al., 2018)	1.383	1.230	76.6	61.0	333.1	2.111
GNOT (Hao et al., 2023)	1.416	1.315	75.3	56.4	338.7	1.606
GNO (Kovachki et al., 2023)	1.422	1.244	75.6	57.9	329.8	8.504
Transolver (Wu et al., 2024)	1.492	1.384	74.8	57.9	338.0	2.672
Transolver++ (Luo et al., 2025)	1.624	1.515	73.2	55.6	337.2	2.768
AdaptiGraph (Zhang et al., 2024)	1.711	1.533	75.8	60.4	330.0	33.916
EGNN (Satorras et al., 2021)	1.815	1.547	70.5	50.3	337.3	8.213
GeoTransolver (Adams et al., 2025)	1.876	1.706	70.6	51.6	335.4	0.066
GINOT (Liu et al., 2025)	1.929	1.740	70.5	52.1	338.2	1.110
MGN (Pfaff et al., 2021)	2.052	1.884	66.8	46.1	336.6	7.295
NOMAD (Seidman et al., 2022)	2.310	2.133	65.2	45.5	327.4	0.483
RIGNO (Mousavi et al., 2025)	2.846	2.458	56.2	37.8	329.4	0.295
UPT (Alkin et al., 2024)	2.912	2.500	56.2	36.1	331.0	0.505
Transformer (Vaswani et al., 2017)	3.068	2.957	60.1	25.2	327.9	0.016
MLP	3.158	3.015	59.8	21.6	347.0	0.002
DROP-Op (ours)	**0.776**	**0.722**	**84.5**	**72.0**	326.7	1.337

sualizes the interaction sequence and the prediction errors. Level-wise results and geometry-group sensitivity appear in Appendices C.1 and B.2.4.

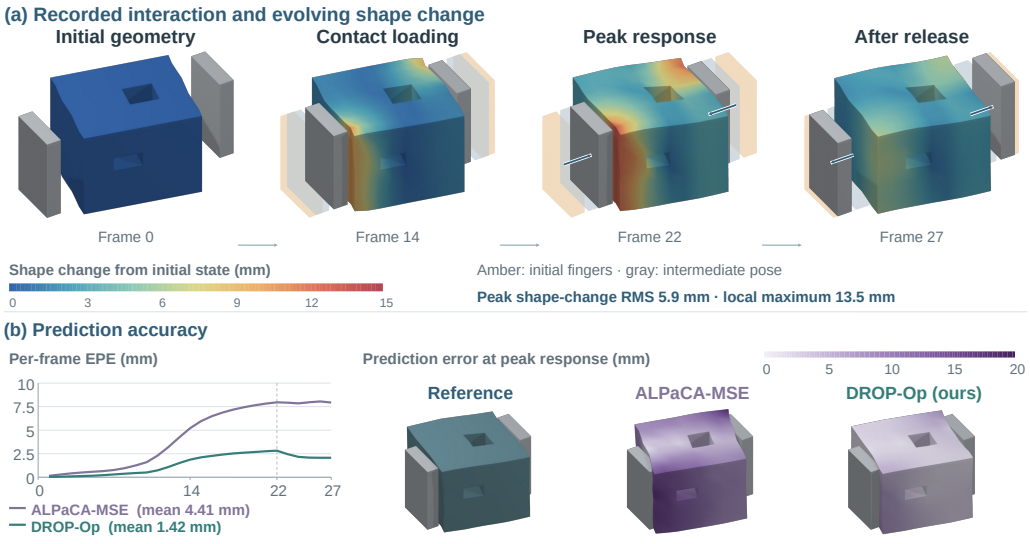

Figure 4: **Interaction and prediction on a two-hole object.** Upper colors show shape change after rigid alignment. Translucent fingers show earlier poses. Lower colors show prediction error at peak deformation on a shared scale. Curves show per-frame EPE; the dashed line marks peak deformation. Selection and rendering: Appendix B.4.

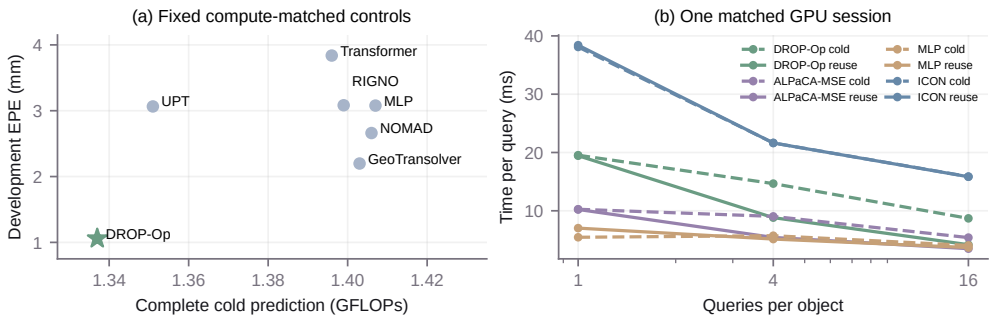

Figure 5: **Prediction accuracy and query cost.** (a) Accuracy at comparable FLOP budgets. (b) Time per query for repeated cold predictions (dashed) and state reuse (solid). Both include identification cost. Protocols are in Appendix C.3.

Table 2: **Object representations for new-action prediction.** A compares representations within a shared predictor. B trains each predictor from scratch. Bold marks the lowest error in each group.

Representation	EPE (mm) ↓	dEPE (mm) ↓
A. Shared prediction features and decoder		
Geometry features	2.509	2.034
Inferred material scalars	2.372	2.009
Observed displacement history	2.042	1.807
Fixed response basis, pooled coefficients	3.330	2.915
Fixed response basis, spatial coefficients	2.792	2.509
Learned global response correction	1.408	1.242
Spatial identification followed by averaging	1.409	1.253
Learned spatial response correction	1.298	1.140
GPR identified state	**1.059**	**0.931**
B. Complete predictors trained from scratch		
Spatial elastic fields	2.261	2.031
Pooled elastic fields	2.272	2.036
GPR response state	**0.907**	**0.844**

A correction adjusts response coefficients. Pooling averages information across locations (Appendix D.1).

At comparable FLOP budgets, DROP-Op reduces EPE and dEPE by 51.8% and 49.8%, respectively, relative to GeoTransolver, the strongest expanded baseline on the development set (Figure 5a). State reuse reduces prediction time (Figure 5b; Section 4.6).

4.3 GPR VERSUS ALTERNATIVE OBJECT REPRESENTATIONS

Table 2A compares geometry, material descriptors and response encodings using fixed prediction features and a shared decoder. GPR has the lowest errors.

In B, each predictor trains from scratch using GPR or elastic fields. We choose elasticity because it governs material deformation during grasping. The field models receive true Young's modulus, Poisson's ratio and element geometry. GPR reduces EPE/dEPE from 2.261/2.031 to 0.907/0.844 mm relative to spatial elastic fields. Both comparisons support its predictive advantage over the tested representations (Appendix D).

4.4 RESPONSE IDENTIFICATION VERSUS CONTEXT CONDITIONING

We compare DROP-Op with context-based predictors and test which parts of its evidence processing explain the advantage (Table 3).

Table 3: **Context-based prediction and DROP-Op mechanism controls.** Errors are in mm (↓). Response-difference error tests matched training examples. A reports development prediction errors, and B reports confirmation errors.

Method	EPE	dEPE	Response difference error
A. Context-based predictors			
IIDA (Evans et al., 2022)	3.878	3.357	0.793
DROP-Op with response attention	1.605	1.406	0.488
ALPaCA-MSE (Harrison et al., 2018)	1.430	1.240	0.373
DROP-Op	**1.057**	**0.920**	**0.292**
B. DROP-Op mechanism controls			
DROP-Op with response attention	1.293	1.174	—
DROP-Op with pooled evidence	0.958	0.897	—
DROP-Op (spatial identification)	**0.776**	**0.722**	—

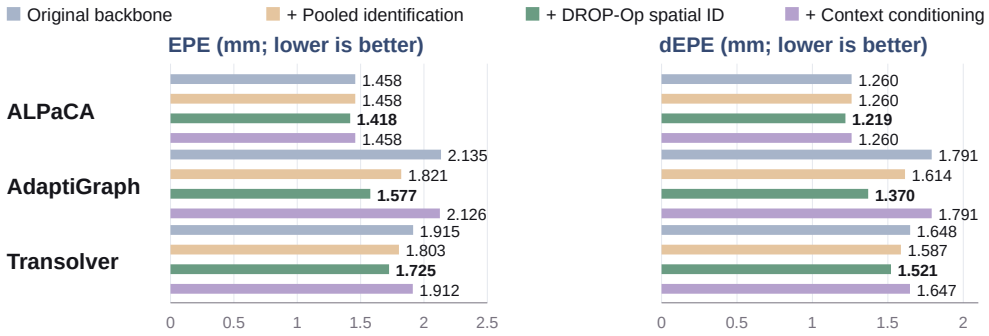

Figure 6: **DROP-Op identification transferred across backbones.** ID denotes response identification. Each backbone is compared with pooled identification, spatial identification and learned context conditioning (Appendix F.1).

Using observed responses. For matched training pairs, only the observed responses differ. Response-difference error measures how well the predicted difference between objects matches the true difference. DROP-Op has lower prediction and response-difference errors than IIDA, ALPaCA-MSE and the attention control (A).

Identification and spatial organization. We train variants with attention replacing identification or spatial evidence pooled before fitting. The attention variant responds to changes in its observations, yet DROP-Op reduces EPE/dEPE by 40.0%/38.5%. Pooling raises EPE from 0.776 to 0.958 mm (B). Both fitting response coefficients and retaining local evidence improve prediction over the tested alternatives (Appendix E).

4.5 Transferring the DROP-Op method across backbones and tasks

Other backbones on our grasping benchmark. We transfer DROP-Op's spatial response-identification mechanism to frozen ALPaCA, AdaptiGraph and Transolver backbones. It outperforms pooled identification and learned conditioning on average under matched evidence and decoders (Figure 6). AdaptiGraph's EPE/dEPE improve by 26.2%/23.5%, and the adapted model beats DROP-Op in nine conditions per metric, providing complementary domain strengths.

External motion and field-prediction tasks. We adapt response identification, denoted DROP-ID, to two tasks with different dynamics (Table 4).

NeRD predicts Cartpole motion under control inputs (Xu et al., 2025). We test changes in gravity and mass/inertia. A correction fitted from observed responses in a learned basis reduces error by 19.75%.

Table 4: **DROP-Op identification on external tasks.** A adds a residual to NeRD (100-step state error, wrapped angles). B replaces DISCO's generator composition (relative field error). Gray highlights learned-basis and generator-composition results.

A. Cartpole: NeRD (Xu et al., 2025)	State $L_2 \downarrow$
Original NeRD	1.4048
NeRD + ridge residual (control)	1.2150
NeRD + DROP-ID (random basis)	1.2696
NeRD + DROP-ID (learned basis)	**1.1274**
Additional result with a task-specific basis	
NeRD + DROP-ID (handcrafted basis)	0.3102
B. Advection–diffusion: DISCO (Serrano et al., 2026)	Relative $L_2 \downarrow$
Native DISCO splitting	0.1792
Latent optimization (Adam)	0.1701
Unit-weight composition (RK4)	0.1860
DROP-ID composition (RK4)	**0.0557**

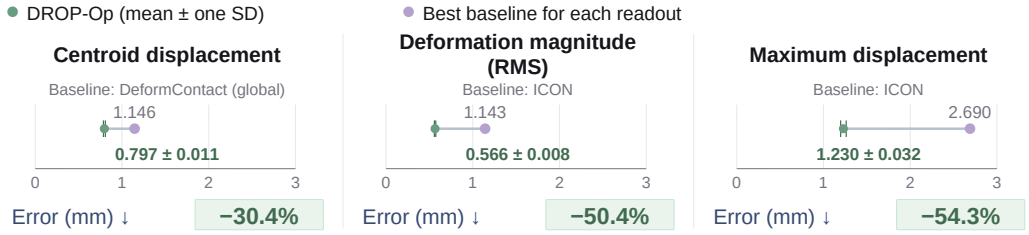

Figure 7: **Readout errors from a shared GPR state.** Each readout is compared with the best baseline in this readout study. Error bars show one standard deviation, and boxes show relative error reductions. RMS denotes root mean square. Protocols are in Appendix G.

DISCO predicts advection–diffusion fields (Serrano et al., 2026). We replace its splitting scheme by fitting weights for frozen dynamics components to observed frames, then predict future frames with RK4. This outperforms native splitting and unit-weight RK4 composition (Appendix F).

4.6 Reusing identified GPR states across actions and readouts

One identified state supports new actions and multiple readouts without retraining. Its centroid, deformation and maximum-displacement errors are lower than the best compared baseline for each readout (Figure 7).

The same predictions support action selection for different motion goals and deformation penalties. DROP-Op reduces regret, the excess cost over the best available action, by 38.2–44.9% (Appendix G.3). At 16 queries, reuse lowers time from 8.702 to 4.187 ms/query relative to repeated cold predictions (Figure 5b).

5 Discussion and limitations

Object responses offer a way to represent combined physical effects without separately estimating each property. Our results show that spatial response identification improves prediction, supports state reuse and gives other backbones complementary strengths. These benefits depend on informative observations and a learned basis that covers the relevant dynamics. Reuse requires compatible object states and interaction conditions. Changes in tools, materials or loading history may require new observations and re-identification (Appendices H and F.4). Future work could improve interaction selection, response coverage and spatial processing efficiency.

AI USE STATEMENT

The authors developed the conceptual framework, mathematical claims, and hypotheses, and led the experimental design. OpenAI Codex assisted in refining these ideas, developing mathematical derivations and proofs, and refining the experiments. Code for method implementation was primarily written by Codex. The authors designed the dataset, while Codex wrote code for data generation and processing. AI assistance was also used for literature search, analysis and interpretation of experimental results, manuscript drafting and revision, and figure and table preparation. The authors reviewed and checked the AI-assisted work and take responsibility for the final content, including the methods, proofs, references, results, and supporting artifacts.

REPRODUCIBILITY STATEMENT

Section 3 and Appendix A describe GPR, DROP-Op, the mathematical derivations, and implementation details. Appendix B documents benchmark generation, data splits, observation adapters, training settings, and evaluation metrics. Appendices C–H provide the protocols and additional results for generalization, representation comparisons, mechanism controls, transfer, reuse, and application conditions. Appendix C.1 distinguishes the original frozen confirmation from subsequent supplementary evaluations. This submission includes the manuscript and appendices, but not source code, model weights, or raw datasets.

REFERENCES

C. Adams, R. Ranade, R. Cherukuri, and S. Choudhry. GeoTransolver: Learning Physics on Irregular Domains Using Multi-scale Geometry Aware Physics Attention Transformer. arXiv preprint arXiv:2512.20399, 2025. URL https://arxiv.org/abs/2512.20399. Revised 2026.

B. Alkin, A. Fürst, S. Schmid, L. Gruber, M. Holzleitner, and J. Brandstetter. Universal Physics Transformers: A Framework For Efficiently Scaling Neural Operators. In *NeurIPS*, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/2cd36d327f33d47b372d4711edd08de0-Abstract-Conference.html.

D. Bear et al. Physion: Evaluating Physical Prediction from Vision in Humans and Machines. In *NeurIPS Datasets and Benchmarks*, 2021. URL https://datasets-benchmarks-proceedings.neurips.cc/paper_files/paper/2021/hash/d09bf41544a3365a46c9077ebb5e35c3-Abstract-round1.html.

L. Bertinetto, J. F. Henriques, P. H. S. Torr, and A. Vedaldi. Meta-learning with differentiable closed-form solvers. In *ICLR*, 2019. URL https://arxiv.org/abs/1805.08136.

Q. Chen, T. Gao, C. Jiang, and T. Wu. EqCollide: Equivariant and Collision-Aware Deformable Objects Neural Simulator. In *KDD*, 2026. URL https://arxiv.org/abs/2506.05797.

R. Dagli et al. VoMP: Predicting Volumetric Mechanical Property Fields. In *ICLR*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/4ea4a1ea4d9ff273688c8e92bd087112-Abstract-Conference.html.

T. Daniel, C. Qi, D. Haramati, A. Zadeh, C. Li, A. Tamar, D. Pathak, and D. Held. Latent Particle World Models: Self-supervised Object-centric Stochastic Dynamics Modeling. In *ICLR*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/cf7ba4b2d14e0f6a0e8247af77745094-Abstract-Conference.html.

B. Evans, A. Thankaraj, and L. Pinto. Context is Everything: Implicit Identification for Dynamics Adaptation. In *ICRA*, 2022. URL https://arxiv.org/abs/2203.05549.

Z. Feng, S. Tao, H. Sun, S. Chen, Z. Zhu, and Y. Liu. MAVEN: A Mesh-Aware Volumetric Encoding Network for Simulating 3D Flexible Deformation. In *ICLR*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/6920c76183516fb8efff75416393a577-Abstract-Conference.html.

R. Gao, Y. Dou, H. Li, T. Agarwal, J. Bohg, Y. Li, L. Fei-Fei, and J. Wu. The ObjectFolder Benchmark: Multisensory Learning with Neural and Real Objects. In *CVPR*, 2023. URL https://openaccess.thecvf.com/content/CVPR2023/html/Gao_The_ObjectFolder_Benchmark_Multisensory_Learning_With_Neural_and_Real_Objects_CVPR_2023_paper.html.

M. Garnelo et al. Conditional Neural Processes. In *ICML*, 2018. URL https://proceedings.mlr.press/v80/garnelo18a.html.

Z. Hao et al. GNOT: A General Neural Operator Transformer for Operator Learning. In *ICML*, 2023. URL https://proceedings.mlr.press/v202/hao23c.html.

J. Harrison, A. Sharma, and M. Pavone. Meta-Learning Priors for Efficient Online Bayesian Regression. In *WAFR*, 2018. URL https://arxiv.org/abs/1807.08912.

I. Huang, Y. Narang, R. Bajcsy, F. Ramos, T. Hermans, and D. Fox. DefGraspNets: Grasp Planning on 3D Fields with Graph Neural Nets. In *ICRA*, 2023. URL https://arxiv.org/abs/2303.16138.

H. Kim, A. Mnih, J. Schwarz, M. Garnelo, A. Eslami, D. Rosenbaum, O. Vinyals, and Y. W. Teh. Attentive Neural Processes. In *ICLR*, 2019. URL https://arxiv.org/abs/1901.05761.

M. Kirchmeyer, Y. Yin, J. Dona, N. Baskiotis, A. Rakotomamonjy, and P. Gallinari. Generalizing to New Physical Systems via Context-Informed Dynamics Model. In *ICML*, 2022. URL https://proceedings.mlr.press/v162/kirchmeyer22a.html.

N. Kovachki, Z. Li, B. Liu, K. Azizzadenesheli, K. Bhattacharya, A. Stuart, and A. 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.

S. Li, K. Wu, C. Zhang, and Y. Zhu. I-PHYRE: Interactive Physical Reasoning. In *ICLR*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/78834433edc3291f4c6cbbd2759324db-Abstract-Conference.html.

S. Li, T. Marwah, J. Shen, W. Sun, A. Risteski, Y. Yang, and A. Talwalkar. CodePDE: An Inference Framework for LLM-driven PDE Solver Generation. *Transactions on Machine Learning Research*, 2026a. URL https://openreview.net/forum?id=eG3Qy5Oux6.

S. Li, R. Shen, Y. Tao, C. Zhang, and Y. Zhu. Neural Force Field: Few-shot Learning of Generalized Physical Reasoning. In *ICLR*, 2026b. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/ed2e1bfa8721d7cb04b8f6f92ab8c0ba-Abstract-Conference.html.

Z. Li, N. Kovachki, K. Azizzadenesheli, B. Liu, K. Bhattacharya, A. Stuart, and A. Anandkumar. Fourier Neural Operator for Parametric Partial Differential Equations. In *ICLR*, 2021. URL https://openreview.net/forum?id=c8P9NQVtmnO.

J. Liang, S. Saxena, and O. Kroemer. Learning Active Task-Oriented Exploration Policies for Bridging the Sim-to-Real Gap. In *Robotics: Science and Systems*, 2020. URL https://arxiv.org/abs/2006.01952.

J. Liang, W. Huang, F. Xia, P. Xu, K. Hausman, B. Ichter, P. Florence, and A. Zeng. Code as Policies: Language Model Programs for Embodied Control. In *ICRA*, 2023. URL https://doi.org/10.1109/ICRA48891.2023.10160591.

M. L. Littman, R. S. Sutton, and S. Singh. Predictive Representations of State. In *NeurIPS*, 2001. URL https://proceedings.neurips.cc/paper_files/paper/2001/hash/1e4d36177d71bbb3558e43af9577d70e-Abstract.html.

Q. Liu, W. Zhong, H. Meidani, D. Abueidda, S. Koric, and P. Geubelle. Geometry-Informed Neural Operator Transformer. arXiv preprint arXiv:2504.19452, 2025. URL https://arxiv.org/abs/2504.19452.

L. Lu, P. Jin, G. Pang, Z. Zhang, and G. E. Karniadakis. Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators. *Nature Machine Intelligence*, 3:218–229, 2021. URL https://doi.org/10.1038/s42256-021-00302-5.

H. Luo, H. Wu, H. Zhou, L. Xing, Y. Di, J. Wang, and M. Long. Transolver++: An Accurate Neural Solver for PDEs on Million-Scale Geometries. In *ICML*, 2025. URL https://proceedings.mlr.press/v267/luo25o.html.

M. Memmel, A. Wagenmaker, C. Zhu, D. Fox, and A. Gupta. ASID: Active Exploration for System Identification in Robotic Manipulation. In *ICLR*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/ae8afec5dabb07091d8f85128250f2ff-Abstract-Conference.html.

S. Mousavi, S. Wen, L. Lingsch, M. Herde, B. Raonić, and S. Mishra. RIGNO: A Graph-based framework for robust and accurate operator learning for PDEs on arbitrary domains. In *NeurIPS*, 2025. URL https://papers.nips.cc/paper_files/paper/2025/hash/dcb91f43033bb1d367d1848806dee98d-Abstract-Conference.html.

T. Pfaff, M. Fortunato, A. Sanchez-Gonzalez, and P. W. Battaglia. Learning Mesh-Based Simulation with Graph Networks. In *ICLR*, 2021. URL https://openreview.net/forum?id=roNqYL0_XP.

M. Saleh, M. Sommersperger, N. Navab, and F. Tombari. Physics-Encoded Graph Neural Networks for Deformation Prediction under Contact. In *ICRA*, 2024. URL https://arxiv.org/abs/2402.03466.

V. Garcia Satorras, E. Hoogeboom, and M. Welling. E(n) Equivariant Graph Neural Networks. In *ICML*, 2021. URL https://proceedings.mlr.press/v139/satorras21a.html.

J. H. Seidman, G. Kissas, P. Perdikaris, and G. J. Pappas. NOMAD: Nonlinear Manifold Decoders for Operator Learning. In *NeurIPS*, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/24f49b2ad9fbe65eefbfd99d6f6c3fd2-Abstract-Conference.html.

L. Serrano, J. Han, E. Oyallon, S. Ho, and R. Morel. Test-time Generalization for Physics through Neural Operator Splitting. arXiv preprint arXiv:2602.00884, 2026. URL https://arxiv.org/abs/2602.00884.

B. Shen, Z. Jiang, C. Choy, L. J. Guibas, S. Savarese, A. Anandkumar, and Y. Zhu. ACID: Action-Conditional Implicit Visual Dynamics for Deformable Object Manipulation. In *Robotics: Science and Systems*, 2022. URL https://arxiv.org/abs/2203.06856.

M. Takamoto, T. Praditia, R. Leiteritz, D. MacKinlay, F. Alesiani, D. Pflüger, and M. Niepert. PDEBench: An Extensive Benchmark for Scientific Machine Learning. In *NeurIPS Datasets and Benchmarks*, 2022. URL https://arxiv.org/abs/2210.07182.

E. Todorov, T. Erez, and Y. Tassa. MuJoCo: A physics engine for model-based control. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5026–5033, 2012. URL https://doi.org/10.1109/IROS.2012.6386109.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention Is All You Need. In *NeurIPS*, 2017. URL https://papers.nips.cc/paper_files/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

J. C. Willems. Dissipative Dynamical Systems Part I: General Theory. *Archive for Rational Mechanics and Analysis*, 45:321–351, 1972. URL https://doi.org/10.1007/BF00276493.

H. Wu, H. Luo, H. Wang, J. Wang, and M. Long. Transolver: A Fast Transformer Solver for PDEs on General Geometries. In *ICML*, 2024. URL https://proceedings.mlr.press/v235/wu24r.html.

J. Xu, E. Heiden, I. Akinola, D. Fox, M. Macklin, and Y. Narang. Neural Robot Dynamics. In *CoRL*, 2025. URL https://proceedings.mlr.press/v305/xu25g.html.

Z. Xu, J. Wu, A. Zeng, J. B. Tenenbaum, and S. Song. DensePhysNet: Learning Dense Physical Object Representations via Multi-step Dynamic Interactions. In *Robotics: Science and Systems*, 2019. URL https://www.roboticsproceedings.org/rss15/p46.pdf.

L. Yang, S. Liu, T. Meng, and S. J. Osher. In-Context Operator Learning with Data Prompts for Differential Equation Problems. *Proceedings of the National Academy of Sciences*, 120(39): e2310142120, 2023. URL https://doi.org/10.1073/pnas.2310142120.

A. J. Zhai, Y. Shen, E. Y. Chen, G. X. Wang, X. Wang, S. Wang, K. Guan, and S. Wang. Physical Property Understanding from Language-Embedded Feature Fields. In *CVPR*, 2024. URL https://arxiv.org/abs/2404.04242.

K. Zhang, B. Li, K. Hauser, and Y. Li. AdaptiGraph: Material-Adaptive Graph-Based Neural Dynamics for Robotic Manipulation. In *Robotics: Science and Systems*, 2024. URL https://www.roboticsproceedings.org/rss20/p010.pdf.

APPENDICES

The appendices follow the main text. Appendix A gives method details and cost; Appendix B defines the benchmark and evaluation; Appendix C reports additional generalization and efficiency results. Appendices D–E detail representation and mechanism comparisons, followed by transfer (F), reuse (G) and application conditions (H).

A DROP-OP: DERIVATIONS, IMPLEMENTATION AND COMPUTATION

This section supplements Section 3 with the identification derivation, implementation details and full prediction cost.

A.1 OPERATOR FAMILIES CONSIDERED IN THIS STUDY

Figures 8 and 9 summarize the methods considered in our AI-assisted exploration. The groups highlight overlapping design choices: direct prediction, spatial processing, context conditioning, identification and composition. AI assistance supported implementation and analysis, as described in the AI use statement.

A.2 RESPONSE DEFINITION AND OBSERVATIONS

For object o with reference domain Ω_o, initial state s_0 and interaction interface b, an action $u \in \mathcal{U}$ induces motion χ_o. The displacement and GPR response family are

$$d_o(x,t;u) = \chi_o(x,t;s_0,b,u) - x, \qquad \mathcal{G}_o^{s_0,b,\mathcal{U}} = \{(u, d_o(\cdot,\cdot;u)) : u \in \mathcal{U}\}. \tag{5}$$

Writing $\mathcal{G}_o[u] = d_o(\cdot,\cdot;u)$, the sampling operator Π_k selects the observed positions and times:

$$v_k = \Pi_k \mathcal{G}_o[u_k], \qquad \mathcal{C}_o = \{(u_k, \Pi_k, v_k)\}_{k=1}^K. \tag{6}$$

The main operator identifies a finite state from g_o and $\mathcal{C}_o$, then queries it under new actions.

Which task outputs the observations determine. For fixed noiseless linear evidence $v = B\eta$ and task output $q = Q\eta$, the output is determined by the evidence precisely when

$$B\eta_1 = B\eta_2 \Longrightarrow Q\eta_1 = Q\eta_2 \quad \Longleftrightarrow \quad \ker B \subseteq \ker Q. \tag{7}$$

If the kernel inclusion holds, identical evidence implies an output difference of zero. Otherwise, an $h \in \ker B \setminus \ker Q$ gives identical evidence for η and $\eta + h$ but different outputs. Equivalently, $Q = LB$ for some L, so $q = Lv$. Multiple tasks require $\ker B \subseteq \bigcap_j \ker Q_j$. Recovering every physical parameter requires the stronger condition $\ker B = \{0\}$. For $B = [1\ 1]$, the sum is identifiable but the difference is not. With observation error, $\|\delta q\| \le \|L\|\|\delta v\|$ also depends on numerical conditioning.

For a sampled displacement field $D \in \mathbb{R}^{N\times 3}$, define $c(D) = D^\top e_N/N$ and $P_c = I - e_N e_N^\top/N$. Then $D = e_N c(D)^\top + P_c D$ separates mean motion and centered deformation within the same response representation.

A.3 SPATIAL RESPONSE IDENTIFICATION

Geometry features $h_g = \mathcal{E}_\theta(g_o)$ and action u produce a shared prediction and rank-12 basis, $(p^u, b^u) = \mathcal{F}_\theta(h_g, u)$. With initial-frame differences $\Delta p = p_t - p_0$, $\Delta b = b_t - b_0$ and RMS radius r, four interactions with four observed frames each give, at anchor i,

$$(X_i)_{(k,\ell):} = (\Delta b_{t\ell i}^{u_k})^\top, \qquad (E_i)_{(k,\ell):} = r^{-1} d_{t\ell i}^{u_k} - \Delta p_{t\ell i}^{u_k}. \tag{8}$$

The compact notation in Eq. 2 absorbs physical scaling and initial-frame subtraction: $\mu_\theta = r\Delta p$, $\phi_\theta = r\Delta b$, and $a_o(x_n) = \bar{A}_n$. Thus it is the same decoder used below, with zero initial displacement.

Here $X_i \in \mathbb{R}^{16\times 12}$ contains observed-action basis features and $E_i \in \mathbb{R}^{16\times 3}$ contains normalized response residuals. Geometry-based sharing uses $s_{ij} = 0.75\, 1_{i=j} + 0.25 P_{ij}$ with row-normalized

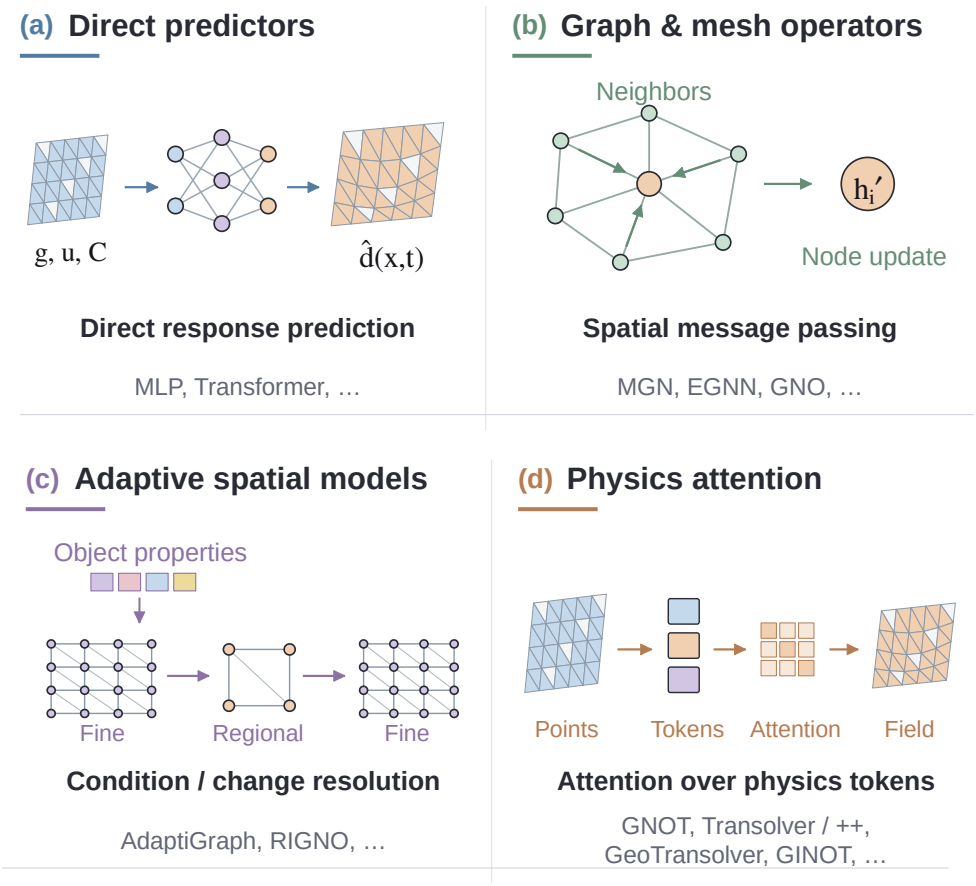

Figure 8: **Operator families: prediction and spatial processing.** (a) Direct predictors. (b) Graph and mesh models (Pfaff et al., 2021; Satorras et al., 2021; Kovachki et al., 2023). (c) Property conditioning and regional computation (Zhang et al., 2024; Mousavi et al., 2025). (d) Attention-based field predictors (Hao et al., 2023; Wu et al., 2024; Luo et al., 2025; Adams et al., 2025; Liu et al., 2025). These panels group related design choices rather than identical architectures. Here g, u and $\mathcal{C}$ denote geometry, an action and observed interactions, respectively.

nonnegative P. Nonnegative diagonal W_j weights observations. The conditional fit is

$$A_i = \arg\min_A \frac{1}{2}\sum_j s_{ij}\|W_j^{1/2}(X_jA - E_j)\|_F^2 + \frac{\lambda_i}{2}\|A - C_i\|_F^2. \tag{9}$$

Equation 3 writes the same objective using stacked weighted evidence. The block for anchor j is $Q_i^{(j)} = \sqrt{s_{ij}}W_j^{1/2}X_j$ and $Y_i^{(j)} = \sqrt{s_{ij}}W_j^{1/2}E_j$. Stacking these blocks gives $Q_i^\top Q_i = G_i$ and $Q_i^\top Y_i = H_i$. Dropping the common factor $1/2$ leaves the minimizer unchanged. The implementation shares these statistics directly.

The data term fits the observations, while the regularizer constrains the fit around C_i, a global ridge estimate plus a response-scaled correction. Both C_i and positive λ_i depend on geometry and response statistics. Define $G_i = \sum_j s_{ij}X_j^\top W_j X_j$ and $H_i = \sum_j s_{ij}X_j^\top W_j E_j$. These are the sufficient statistics in the linear system of Eq. 4. Building on learned-feature regression (Harrison et al., 2018; Bertinetto et al., 2019), DROP-Op combines local response equations, geometry-based sharing and conditional regularization. Its state is $z_o = (h_g, \{(a_i, A_i)\}_{i=1}^{32})$. Geometry weights interpolate coefficients as $\overline{A}_n = \sum_i T_{ni}A_i$ for Eq. 2; task readouts apply $\widehat{q}_j = \rho_j(\widehat{d})$ to the resulting field. This predictive-state view connects response evidence to subsequent queries (Littman et al., 2001; Garnelo et al., 2018; Yang et al., 2023).

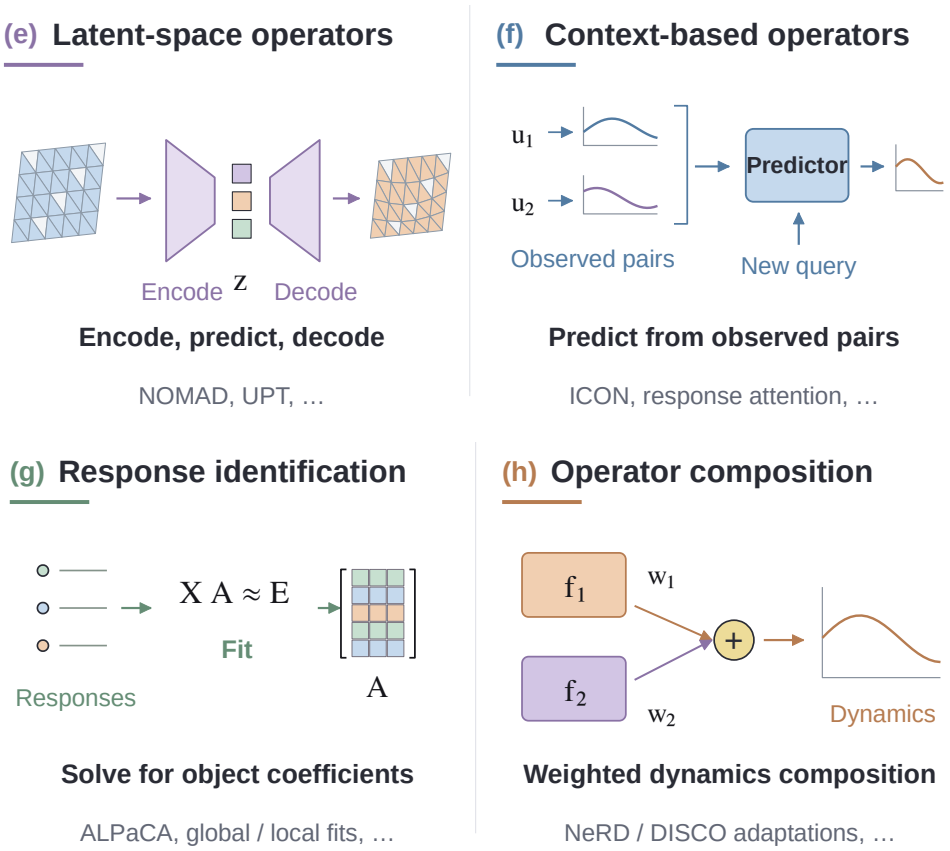

Figure 9: **Operator families: latent prediction, context and adaptation.** (e) Encoding and decoding a latent state z (Seidman et al., 2022; Alkin et al., 2024). (f) Prediction from observed input–response pairs (Yang et al., 2023). (g) Fitting response coefficients, illustrated by $XA \approx E$ (Harrison et al., 2018). (h) Combining dynamics components. Our NeRD adaptation adds a residual correction, while our DISCO adaptation replaces composition weights (Appendix F). Models may use more than one of these mechanisms.

Conditional solve and learned regularization. Holding the evidence-dependent quantities in Eq. 9 fixed gives gradient $(G_i + \lambda_i I)A - H_i - \lambda_i C_i$. Nonnegative weights imply $G_i \succeq 0$, and $\lambda_i > 0$ makes the Hessian positive definite. Equation 4 therefore has the unique conditional solution $A_i = (G_i + \lambda_i I)^{-1}(H_i + \lambda_i C_i)$. The implementation uses

$$\lambda_i = \big(0.1\ \mathrm{tr}(G_i)/12 + 10^{-4}\big) \exp\big((\log 4)\tanh s_i\big),$$
$$C_i = C_g + \alpha_i D_i, \qquad \alpha_i = \frac{e_i}{e_i + \langle D_i, G_i D_i\rangle + 10^{-8}}. \tag{10}$$

Here C_g is a global ridge estimate, e_i is the shared squared normalized-response magnitude, and s_i, D_i are learned functions of response statistics. Each anchor has one positive scalar λ_i and $0 \le \alpha_i < 1$. New evidence changes the center and regularizer as well as the normal equations.

Training gradients and evidence updates. Training minimizes physical-unit query MSE,

$$\mathcal{L} = \frac{1}{3|\mathcal{Q}|} \sum_{(o,u,t,n)\in\mathcal{Q}} \|\widehat{d}_{o,u,t,n} - d_{o,u,t,n}\|_2^2. \tag{11}$$

For $M_i = G_i + \lambda_i I$ and $B_i = H_i + \lambda_i C_i$, differentiating $M_i A_i = B_i$ gives

$$\mathrm{d}A_i = M_i^{-1}[\mathrm{d}H_i + \lambda_i \mathrm{d}C_i + (C_i - A_i)\mathrm{d}\lambda_i - (\mathrm{d}G_i)A_i]. \tag{12}$$

An adjoint solve $M_i^\top V_i = \partial\mathcal{L}/\partial A_i$ yields $\mathrm{d}\mathcal{L} = \langle V_i, \mathrm{d}B_i - (\mathrm{d}M_i)A_i\rangle_F$. Thus gradients include changes in features, evidence statistics and regularization.

For $Ma = h + \Lambda C$ and $M'a' = h' + \Lambda' C'$, subtract $M'a$ from the second equation to obtain

$$a' - a = M'^{-1}(\Delta h - \Delta G\, a) + M'^{-1}[\Delta\Lambda(C - a) + \Lambda'\Delta C]. \tag{13}$$

The implementation solves this system. With geometry and query features fixed, it also gives the corresponding prediction update. This is an algebraic decomposition; component norms need not add.

A.4 Implementation and reproducibility

The global center C_g uses spatially averaged normal equations. The quantity e_i is the shared sum of squared normalized observed displacements.

The query decoder combines a full-resolution 96-site direct path with two 32-site updates that reuse the geometry-block weights. A 192-parameter action adapter uses bounded differences of known-command embeddings at lags 1, 4 and 8. Geometry weights interpolate the identified coefficients as in Eq. 2. Absolute positions add the initial coordinates. Network parameters are fixed during identification and query reuse. Core checks verify model-state and geometry compatibility; the composition interface in Appendix G.1 also specifies tools, units and observation provenance.

A.5 Full operator computation: from observations to task outputs

The cost includes geometry, command and response features, identification, all output sites and state validation. Counts refer to the compact main model.

For N sites, T frames, K observed interactions and J observed frames per interaction, the complete cost separates identification from querying:

$$\underbrace{C_{\mathrm{cold}}}_{C_{\mathrm{id}}+C_{\mathrm{query}}} = \underbrace{G(N) + KA(T) + F(KJ) + R}_{C_{\mathrm{id}}} + \underbrace{A(T) + F(T) + D}_{C_{\mathrm{query}}}. \tag{14}$$

Here G, A, F, R, D denote geometry processing, command processing, response-field features, identification and decoding. The following ledger expands these terms, including algebra, packing and interpolation. Average cost over Q compatible queries is $C_{\mathrm{query}} + C_{\mathrm{id}}/Q$.

A.5.1 Dimensions, geometry and command processing

Let N be object sites, M spatial anchors, T frames, G commanded gripper points, K acquired interactions, J observed frames per interaction, r basis rank and d output coordinates. DROP-Op uses

$$(N, M, T, K, J, r, d) = (96, 32, 28, 4, 4, 12, 3), \quad (w, H, a, p, L) = (96, 354, 64, 64, 2), \tag{15}$$

where w, H are spatial feature/hidden widths, a command width, p prediction-head width and L shared spatial blocks. A multiply–accumulate counts as two operations. The ledger also counts biases and elementwise arithmetic; latency measurements include selection and data movement.

Geometry is encoded once per state. Coordinate lifting, dense neighborhood products and the two $3w \to H \to w$ blocks give leading arithmetic

$$G(N) = 6Nw + 2LN^2w + 8LNwH + G_{\mathrm{rest}}(N, M). \tag{16}$$

The remainder includes normalization, radius, N^2 pair distances, NM transfer weights, M^2 anchor neighborhoods, pooling, activations and residuals. The implementation uses dense matrices, including zero-weight neighborhoods, and therefore retains the N^2 term.

Each command trajectory is processed at all T frames, including each of the K support trajectories. Gripper aggregation uses TMG anchor–gripper pairs; action MLPs map $(w + 20) \to a \to a$. The three command-memory lags process known commands and add computational work:

$$A(T, G) = 2TM[(w + 20)a + a^2] + A_{\mathrm{geom}}(TMG) + A_{\mathrm{memory}}(\ell TMa) + A_{\mathrm{elem}}, \quad \ell = 3. \tag{17}$$

Here $A_{\mathrm{geom}} = O(TMG)$ includes displacement/velocity moments and weights, and $A_{\mathrm{memory}} = O(\ell TMa)$. The support path processes all T command frames, including the J observed response frames.

Table 5: **Complete cold-prediction cost.** Components are disjoint and include identification and the first query.

Component	Cost (GFLOPs)
Geometry encoding	0.056589824
Support command features	0.088700148
Support response field	0.418448896
Support packing/interpolation	0.006558616
Statistics, solves, coefficient transfer	0.000919999
Learned regularizer	0.000146911
New command features	0.022175181
Query response field	0.732285568
Query interpolation/readout	0.011317376
Total	**1.337142519**

A.5.2 Spatial field, identification and final decoding

For U temporal positions, direct full-site features, population heads and basis heads cost $2UN[(w+2a)w+wp+pd+(w+a)p+pr]$. Coarse spatial updates, dense anchor mixing and interpolation give

$$\begin{aligned} F(U) = {} & 2UN[(w+2a)w+wp+pd+(w+a)p+pr] \\ & + 8LUMwH + 2LUM^2w + 2UNM(w+a) + F_{\mathrm{elem}}(U). \end{aligned} \tag{18}$$

The context uses $U = KJ = 16$ and a query uses $U = T = 28$. All $N = 96$ sites are decoded. Parameter sharing saves weights but every block application contributes arithmetic.

Set $S = KJ$. For observed basis X_m and residual Y_m, the sufficient statistics are $X_m^\top W_m X_m$ and $X_m^\top W_m Y_m$. Their dense sharing, the global and local solves, and coefficient transfer cost

$$\begin{aligned} F_{\mathrm{stats}} &= 2MS(r^2+rd), & F_{\mathrm{sharing}} &= 2M^2(r^2+rd), \\ F_{\mathrm{solve}} &= (M+1)\left[\frac{r(r-1)(4r+1)}{6} + d(2r^2-r)\right], & & \\ F_{\mathrm{transfer}} &= 2NMrd, & F_{\mathrm{contract}} &= 2TNrd. \end{aligned} \tag{19}$$

The ledger counts pivoted-LU arithmetic. The 33 rank-12 solves require 62,898 operations. Statistics, solves, transfer and the learned regularizer together use 1,066,910 operations, or 0.080% of the cold total. Packing, masking and output operations are also included.

Combining the paths, including the remaining packing P and output arithmetic D, yields

$$\begin{aligned} C_{\mathrm{id}} = {} & G(N) + KA(T,G) + F(KJ) + P + F_{\mathrm{stats}} + F_{\mathrm{sharing}} \\ & + F_{\mathrm{solve}} + F_{\mathrm{transfer}} + F_{\mathrm{prior}} + F_{\text{other algebra}}, \end{aligned} \tag{20}$$

$$C_{\mathrm{query}} = A(T,G) + F(T) + F_{\mathrm{contract}} + D, \tag{21}$$

$$C_{\mathrm{cold}} = C_{\mathrm{id}} + C_{\mathrm{query}}. \tag{22}$$

The ledger assigns each operation to one component. Two fixed training inputs reproduce the total with unchanged predictions and weights.

A.5.3 Scaling and amortized inference cost

With fixed widths, anchors, rank and observation count, cold cost is $O(N^2 + (KJ+T)N + (K+1)TG)$ and query cost is $O(TN + TG)$. Producing the output requires $\Omega(TNd)$ writes. Dense geometry causes the quadratic cold term.

For Q compatible queries, the state depends on acquired evidence and geometry, not the new command:

$$C(Q) = C_{\mathrm{id}} + QC_{\mathrm{query}} = 0.571364394 + 0.765778125Q \text{ GFLOPs},$$
$$\bar{C}(Q) = 0.765778125 + \frac{0.571364394}{Q}. \tag{23}$$

At $Q = 1, 16$, costs per query are 1.337142519 and 0.801488400 GFLOPs. As Q grows, the cost approaches the full query cost of 0.765778125 GFLOPs. Relative to re-running the entire same operator, Q16 removes 40.06% of arithmetic and the asymptotic saving is 42.73%; these savings are measured relative to repeated cold calls of DROP-Op. Changed evidence, geometry or interface requires valid state maintenance or re-identification. Task readouts add their own $O(TNd)$ reductions in general, or $O(Nd)$ for one selected frame, while sharing the already computed field.

Workspace for geometry includes $O(N^2 + NM + M^2 + Nw)$ tensors; query activations include $O(TN(w+a+r)+TM(H+a)+TMG)$, in addition to weights and library workspace. Algebra adds $O(Mr^2 + Mrd + Nrd)$ storage. These terms describe inference workspace. Training additionally requires gradient and optimizer storage. DROP-Op retains 82.39 KiB in the measured reuse wrapper, which includes geometry and interpolated state rather than coefficients alone.

A cached baseline also has an initial and a per-query cost. Relative speed therefore depends on both terms and the query count; Appendix C.3 measures the complete workload.

B Benchmark and evaluation protocols

This section supplements Section 4.1 with the domain hierarchy, simulation and split settings, observation adapters, metrics and rendering procedure.

B.1 Dataset design and evaluation factors

Table 6 summarizes the hierarchy. The following tables specify its domain tests; Appendix B.2 gives simulation settings, training ranges and split construction.

B.1.1 Versioned tasks and the complete domain hierarchy

V1-A has five K0 conditions: IID, high strain, eccentricity, grasp angle and hard stiffness. V1-B has 24 native K4 panels: eleven domains paired with familiar/new actions and two object-instance checks. V2-L1 extends the domain tests to richer shapes and material layouts and comprises 32 native panels. V2 native evaluation has 72 panels: 32 L1, 16 L2 and 24 V1-source panels. The primary 48-panel development comparison excludes two duplicated topology panels from the 50-panel comparison. Native training protocols remain separate.

The primary comparison contains 1,306 query appearances across the 48 distinct panels defined above. Familiar actions can occur with shifted geometry or materials. Errors average queries within identities and identities within panels.

Table 6: **Benchmark hierarchy and evaluation scope.** K counts observed support interactions.

Setting	Test conditions	Evaluation purpose
V1-A (K0)	Strain, grasp location and angle, stiffness	Prediction without response evidence.
V1-B (K4)	Instance, mechanics, shape and topology; familiar and new actions	Base hierarchy with response evidence.
V2-L1 (K4)	V1-B domains with more complex shapes and material layouts	Greater physical difficulty in the same task hierarchy.
V2-L2 (K4)	Contact, motion rate, actuation, grippers, spatial mechanics and joint shifts	Interaction, interface and combined OOD.

Table 7: **Aligned V1-B and V2-L1 domains.** Each domain pairs familiar and OOD actions. Heldout compositions are training-excluded factor combinations evaluated in development.

Test condition	Prediction task
Seen-identity query IID	Predict another recorded query for a Fit identity.
Object-instance IID	Transfer to another instance/size under the corresponding source protocol.
Single-property interpolation	Interpolate a controlled mechanical factor.
Single-property extrapolation	Extrapolate that factor beyond its training range.
Heldout property composition	Combine mechanical settings not jointly present in training.
Geometry interpolation	Interpolate continuous shape parameters.
Geometry extrapolation	Extrapolate shape parameters.
Geometry-family OOD	Transfer to a different family of shapes.
Geometry–property composition	Combine geometry and mechanics across a missing pairing.
Topology-parameter OOD	Vary a parameter within a topological family.
Topology-class OOD	Transfer across topological families.

Table 8: **Sixteen V2-L2 domains grouped by evaluation purpose.** Each contributes 16 query appearances.

Factor group	Test conditions	Evaluation purpose
Geometry and load paths	Complex geometry family; topology and load path	Structural organization beyond simple shape parameters.
Spatial mechanics	Heterogeneity layout; heterogeneity contrast and correlation	Placement and variation of spatial material response.
Interaction regime	Large deformation; contact-regime switch; deformation rate and hysteresis	Variation in motion rate and contact regime; loading history is not independently controlled.
Actuation and tools	Partial actuation/asymmetry; finger count; rigid-gripper shape; true flexible fingers	Changed drive, contact directions and tool compliance.
Crossed factors	Geometry $\times$ heterogeneity; object $\times$ gripper; topology $\times$ boundary; five-factor joint; rare-tail joint distribution	Whether single-factor capabilities compose under joint shifts.

B.2 Observation adapters, geometry metrics and split semantics

Errors are in mm; overlap scores are reported as percentages. K counts observed interactions and Q counts queries after identification. Fit denotes training; development and frozen confirmation are separate cohorts. A panel is an evaluation condition. Training-seed SD, adapter-initialization SD and resampling intervals are identified where used.

B.2.1 Model inputs and observation adapters

The common observation boundary contains initial coordinates $g \in \mathbb{R}^{96\times 3}$, four support commands sampled at frames $(0, 9, 18, 27)$, support positions $Y \in \mathbb{R}^{4\times 4\times 32\times 3}$ at site indices $(0, 3, \ldots, 93)$, a Boolean availability mask of shape $4 \times 4 \times 32$, and the complete new command $U_* \in \mathbb{R}^{28\times G\times 3}$. G is the number of commanded gripper sites. Query response, material labels and simulator connectivity are excluded. Intermediate support command frames are reconstructed by piecewise linear

Table 9: **How baselines use the common K4 observations.** The descriptions identify where spatial response information is retained or compressed.

Method	How observed responses enter prediction
AdaptiGraph	56 moment values per interaction become a five-value code; codes are averaged across four interactions. Time order is retained before encoding; spatial correspondence is compressed.
ICON	Commands and 128 location–time response tokens per demonstration. Query tokens retain the new command, location and time. The adapted model trains from scratch.
ALPaCA	Global regression in a learned rank-64 basis over 512 geometry/action/time descriptors and observed displacements. Spatial indexing is retained; MSE and NLL are separate arms.
DeformContact	Backbone and identification/residual modules train jointly from scratch. Global/local specifies this module structure; leave-one-out calibration is detached.
DROP-Op	32 local response-equation sets, shared statistics and rank-12 regularized solves. Spatial coefficients are retained for new commands.
Frozen context controls	Mean-pooling or anchor attention over 512 response tokens, with frozen parent features and displacement MSE. These controls use deterministic conditioning.

interpolation between permitted samples; they are not additional measured evidence. Unobserved support-response slots are not imputed as measurements.

All models receive the same raw observations, but their adapters retain different information. The table specifies the resulting bottlenecks in the implementations evaluated here.

The frozen context controls are deterministic conditioning models rather than full probabilistic neural processes.

For the fingerprint, write $d_{ktn} = (y_{ktn} - g_n)/r$ and normalized commanded motion m_{ktj}. Each frame contributes

$$[\mathrm{mean}_n\, d, \mathrm{std}_n\, d, \sqrt{\mathrm{mean}_n\, \|d\|^2}, \max_n \|d\|, \mathrm{mean}_j\, m, \mathrm{std}_j\, m] \in \mathbb{R}^{14}.$$

The four-frame concatenation is passed through $56 \to 2h \to h \to 5$, with GELU and LayerNorm, then averaged over interactions. The original parameter budget determines width h. The reported AdaptiGraph results use this low-dimensional summary adapter. ICON and ALPaCA preserve spatially indexed evidence through their respective mechanisms.

Main fresh-138 models use 1,792 Fit queries, effective batch eight, 128 query point–time samples per query, 224 updates per pass and 30,912 total updates. The shared AdamW recipe uses learning rate $3 \cdot 10^{-4}$, cosine decay to 10^{-5}, epsilon 10^{-8}, weight decay 0.01 and gradient clipping at one. Query displacement MSE is computed in physical units; ALPaCA-NLL is explicitly separate. Query teacher forcing is disabled. These experiments adapt source architectures to the common task, capacity budget and loss.

Jointly trained context adapters. IIDA forms 16 support tokens, one per interaction/frame. Each 449-dimensional token contains all 288 normalized geometry coordinates, a learned 32-dimensional command encoding, normalized time, 96 ordered response coordinates and 32 availability bits. A $449 \to 256 \to 128 \to 64$ ReLU encoder precedes masked averaging. The decoder receives geometry, query site, time, command encoding and this 64-dimensional latent ($388 \to 256 \to 256 \to 3$); it subtracts the same-latent initial-time prediction. Thus spatial ordering enters before averaging, unlike the hand-designed response fingerprint.

The jointly trained direct-response attention control is specified in Appendix E.2. Its response path and the earlier adapter's attenuation are analyzed in Appendix E.1. All controls use deterministic displacement prediction and the common training recipe.

Explicit development-panel mapping. The primary development comparison uses 48 distinct panels: two duplicate topology panels (52/53) are excluded and two legacy K4 instance panels (22/23) are included. All headline development results use this mapping. The alternative V2-only48 diagnostic is not combined with the primary aggregate. Frozen confirmation uses the separate 46-panel cohort.

B.2.2 SURFACE-TUBE IOU AND DEFORMATION-CHANGE IOU

These metrics compare physical-distance tubes around triangulated surfaces. Let $V^0 \in \mathbb{R}^{N_v \times 3}$ be the full rest mesh and $\mathcal{F}$ its boundary triangles. Only the scorer accesses these full-mesh quantities. Explicit stored surface triangles are used when present; otherwise faces occurring exactly once among tetrahedral faces define $\mathcal{F}$.

The predictor outputs positions $\widehat{x}_{tj}$ at 96 tracked vertices $s_j = V^0_{i_j}$. A cubic radial-basis interpolant decodes displacement to every rest vertex:

$$f_t(x) = \sum_{j=1}^{96} a_{tj}\|x - s_j\|^3 + B_t^\top [1, x^\top]^\top, \qquad \widehat{V}_t = V^0 + f_t(V^0).$$

The interpolation equations enforce $f_t(s_j) = \widehat{x}_{tj} - s_j$ and the degree-one polynomial side constraints; smoothing is zero. This is the implemented SciPy `RBFInterpolator` configuration (cubic kernel, degree one). The prediction surface $\widehat{S}_t$ uses vertices $\widehat{V}_t$ with the same triangles $\mathcal{F}$. Truth S_t uses the simulator's full deformed vertices. Decoding the true 96 points provides a separate sampling-oracle diagnostic. Scoring uses the full truth surface.

For a surface S, define the closed unsigned-distance tube

$$T_h(S) = \{x \in \mathbb{R}^3 : \operatorname{dist}(x, S) \le h\}, \quad h \in \{0.002, 0.004, 0.008\}\ \mathrm{m}.$$

Let $A_t = T_h(S_t)$, $\widehat{A}_t = T_h(\widehat{S}_t)$ and $A_0 = T_h(S_0)$. With μ denoting volume and $\triangle$ symmetric difference, the query-level metrics pool volumes over $\mathcal{T} = \{1, 8, 15, 22, 27\}$:

$$\mathrm{IoU}_h = \frac{\sum_{t\in\mathcal{T}} \mu(A_t \cap \widehat{A}_t)}{\sum_{t\in\mathcal{T}} \mu(A_t \cup \widehat{A}_t)}, \tag{24}$$

$$\mathrm{dIoU}_h = \frac{\sum_{t\in\mathcal{T}} \mu((A_t \triangle A_0) \cap (\widehat{A}_t \triangle A_0))}{\sum_{t\in\mathcal{T}} \mu((A_t \triangle A_0) \cup (\widehat{A}_t \triangle A_0))}. \tag{25}$$

dIoU measures changes relative to the initial surface in a shared coordinate frame. Rigid translation contributes to the change region. dEPE instead centers displacement error before taking pointwise norms. The scorer shifts all surfaces and integration points by one common box center only for numerical conditioning; this does not alter either set comparison.

For each frame, a shared axis-aligned box encloses the initial surface, truth, all predictions in that integration cohort and the decoded oracle, padded by $h_{\max} + 10^{-7}$ m. Four independently scrambled Sobol sequences use nested prefixes $2^{16}, \ldots, 2^{22}$. Every model and width uses the same points within a frame/scramble. Each integer intersection/union count is multiplied by box volume divided by sample count *before* pooling frames. The four resulting metric ratios are averaged. Seeds are deterministically derived from cohort, query path, scramble and frame.

The per-query numerical screen applies to both metrics, all widths and all models. Adjacent mean changes over the last three powers must be at most 0.005, and the final four-scramble range must be at most 0.005. Each scramble must also have at least 512 pooled target-change and change-union hits. Zero or insufficient change is unresolved rather than assigned perfect agreement. These checks assess numerical convergence. Statistical uncertainty is evaluated separately. Point-to-triangle distance uses Open3D FP32 geometry after common recentering; volume accumulation is FP64. The geometry acceleration reuses scenes and exact counts without changing samples or definitions.

Numerical checks on a synthetic open triangulated surface verify exact-prediction IoU/dIoU of one, stationary-prediction dIoU of zero and affine RBF reconstruction. Reported overlap scores multiply these ratios by 100 to express percentages. Query ratios are averaged within identity, identities within panel and panels equally; seeds are summarized separately before reported means/intervals.

B.2.3 Physical ranges and the meaning of unseen identity

The accepted Fit manifest contains 448 physical identities and 128 distinct rest-mesh hashes. Its realized metadata ranges include soft modulus 32–54 Pa, stiff modulus 64–248.4 Pa, normalized geometry scale 0.94–1.06, interface offset$[-0.008, 0.008]$ m and complexity coordinate 0.82–1.18. Six material layouts cover halfspace splits, off-axis inclusions, multiple regions, graded interfaces, and soft or stiff neck/bridge regions. Fourteen geometry families occur. These are marginal ranges across the realized configurations. Some metadata, including density and friction, are recorded only for specific subgroups.

Auditing the first recorded trajectory of every Fit identity confirms timestep $1.25 \cdot 10^{-4}$ s, the CG solver, object Poisson ratio 0.25, pad friction approximately 0.01, and recorded CCD settings (tolerance 10^{-10}, 500 iterations) in all 448 cases. These statements describe recorded Fit settings; changed-interface evaluation can have additional domain-specific settings. All 3,584 Fit support/query trajectories record action metadata, including close/hold/release fractions, angle, eccentricity, tangential motion, normalized strain and executed strain. Audit metadata never becomes a predictor input.

The common action-support contract restricts Fit angles to ± 7.5 degrees and eccentricity magnitude to 0.06; L1 action-OOD includes± 10 degrees or eccentricity-0.08. L2 extreme-action exclusions include angle magnitude at least 25 degrees, eccentricity magnitude at least 0.18 and the separately defined large-strain coordinate. Normalized and executed strain differ because the geometry-specific safe envelope constrains the commanded deformation.

The completed confirmation has 46 distinct panels: 44 evaluate new physical identities and two evaluate new queries for seen identities. A metadata-only audit finds 460 identity–panel appearances and 284 unique identity strings. Of 380 appearances with explicit rest-mesh hashes, 20 match Fit meshes; those 20 are also seen Fit identities. The remaining 80 appearances lack that hash field but retain mesh-generation recipes; their exact mesh identity remains *unknown*. The exact-mesh check leaves geometric ancestry and independence of generation unresolved. Identity strings, exact rest meshes, continuous shape parameters and geometry families are distinct grouping levels.

The paired bootstrap averages seeds and resamples identity IDs consistently across roles and domains. Identity ID is its clustering unit; geometric ancestry is incompletely recorded. The intervals condition on this identity grouping and the fixed panel design. The following sensitivity analysis groups the saved scores at additional metadata levels while preserving the frozen methods and panel design.

B.2.4 Geometry-group sensitivity

Table 10 summarizes provenance coverage and a post-hoc paired analysis of saved query errors. Groupings preserve repeated identities across panels: 284 identity groups, 278 known-mesh groups with unknown identities separate, 199 groups when unknown meshes share one group, and 17 geometry families. The two unknown-mesh treatments test alternative dependence assumptions. Recorded recipes establish provenance but do not recover unrecorded exact meshes or complete geometric ancestry.

For each of 5,000 draws, every cluster receives one independent unit-mean exponential weight shared across methods, identities and panels. We average queries within identity and the three seeds first, normalize weighted identity means within each panel, then average all 46 panels equally. Positive weights retain every panel. The reported 2.5th–97.5th percentile ranges are descriptive grouping-sensitivity intervals, separate from the original identity-bootstrap intervals. Both error advantages remain positive for all three comparators under all four groupings.

Table 10: **Geometry provenance and grouping sensitivity.** A counts identity–panel appearances unless specified. B reports comparator minus DROP-Op error across 17 geometry families; positive values favor DROP-Op. Brackets are 95% descriptive sensitivity intervals. Results average three seeds.

A. Metadata coverage	
Identity–panel appearances / unique identities	460 / 284
Appearances with an explicit rest-mesh hash	380
Appearances without a hash, with a recorded generation recipe	80
Appearances with a recorded geometry family	460

B. Comparator error minus DROP-Op error		
Comparator	EPE reduction (mm) ↑	dEPE reduction (mm) ↑
ICON	0.420 [0.402, 0.444]	0.338 [0.321, 0.351]
Pooled identification	0.183 [0.169, 0.192]	0.175 [0.161, 0.185]
ALPaCA-MSE	0.358 [0.325, 0.375]	0.322 [0.289, 0.342]



Figure 10: **Gripper configuration.** Training and ID grasps use two opposing fingers; the OOD example uses three. Arrows indicate finger motion; translucent fingers show initial poses.

B.3 Visual comparisons of ID and OOD interactions

Figures 10–14 expand the factor groups introduced in Figure 3. Each shows a recorded training interaction, a new in-distribution (ID) interaction on the same training object, and an out-of-distribution (OOD) example from the corresponding development panel. Main views show frame 14 with a shared camera and physical scale within each figure. Translucent fingers mark frame 0 and arrows indicate recorded motion. Lower insets enlarge structural details. These examples illustrate the registered splits; other physical settings can also differ between their objects.

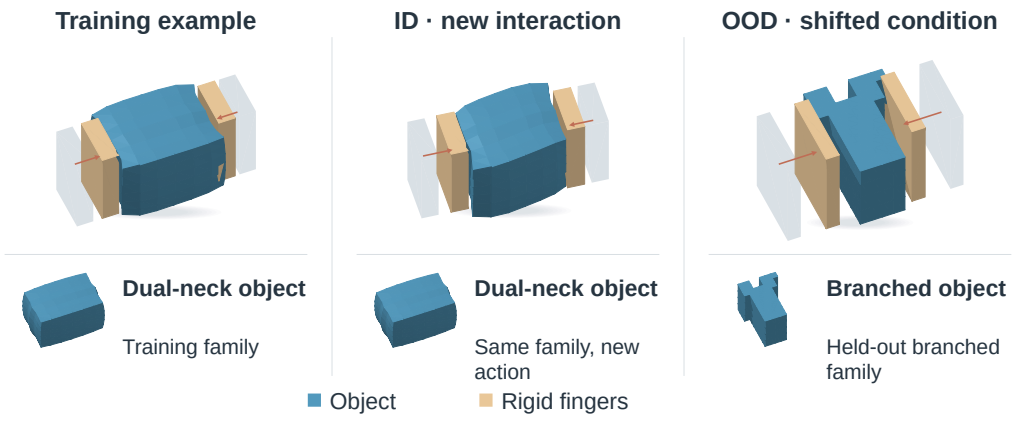

Figure 11: **Topology and load paths.** A training-family object is contrasted with a branched OOD object. Rest-shape insets show the structural difference without gripper occlusion.

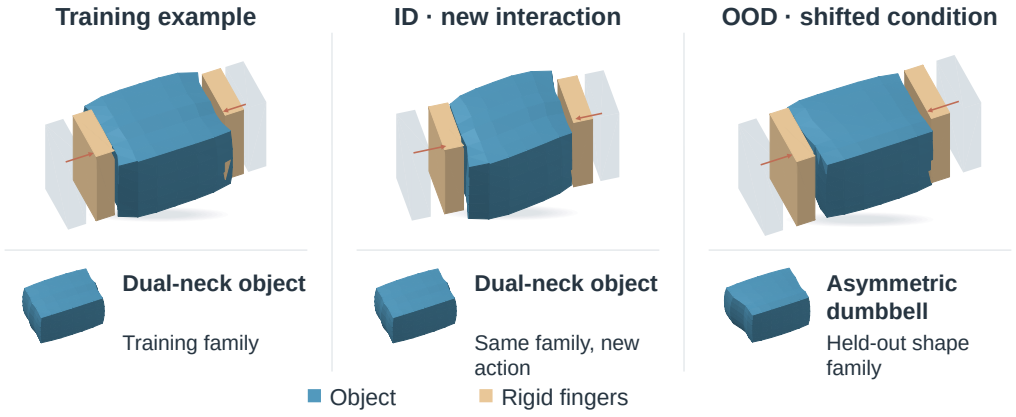

Figure 12: **Geometric generalization.** Training and ID examples from the dual-neck family are contrasted with an OOD example from the asymmetric-dumbbell family. Insets show the recorded rest geometry.

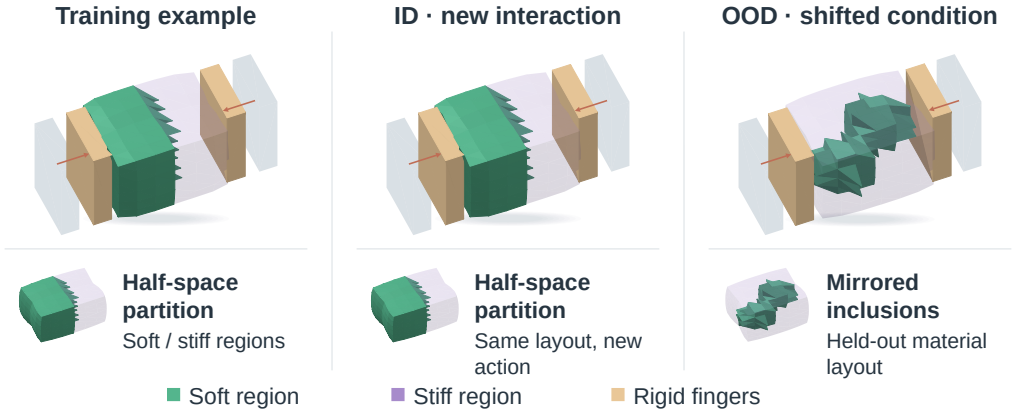

Figure 13: **Spatial material layout.** A half-space partition is contrasted with mirrored soft inclusions. Green denotes soft material and translucent purple denotes stiff material. The partitions match the archived simulation assignments.

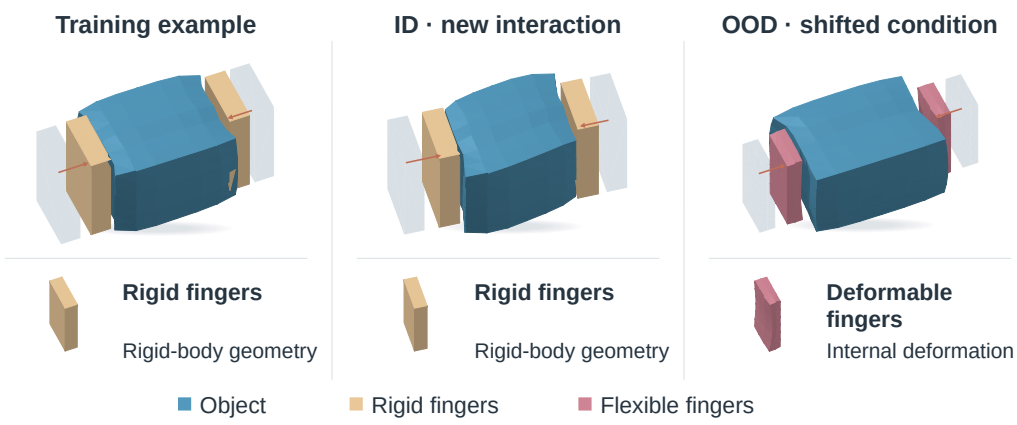

Figure 14: **Deformable gripper fingers.** Tan rigid-body fingers are contrasted with pink deformable fingers. Insets isolate a finger at frame 14 to show its recorded shape.

B.4 PREDICTION EXAMPLES AND SURFACE RENDERING

Figure 4 shows a development trajectory from the contact-regime-shift panel, with two orthogonal, non-intersecting through-holes. The example was selected from 256 exposed Level-2 queries for its visible topology and prediction advantage. It is query position 8, corresponding to action variant 6. Both methods use seed-42 checkpoints trained from scratch for 138 epochs. Full-trajectory ALPaCA-MSE/DROP-Op EPE is 4.4119/1.4246 mm and dEPE is 4.3084/1.3803 mm. Peak-frame EPE is 7.9566/2.8194 mm.

Frames 0, 14, 22 and 27 share a camera and scale; deformation is unscaled. Upper surfaces use the recorded dense geometry, colored by displacement after rigid alignment to the initial shape, on a 0–15 mm scale. Translucent amber and gray fingers mark initial and intermediate poses; arrows indicate recorded motion. Lower reference and prediction surfaces share a reconstruction from the 96 sites. Four nearest rest-configuration sites interpolate each mesh vertex's displacement using inverse-squared-distance weights, reproducing coincident sites exactly. Surface colors interpolate site errors on a common 0–20 mm scale, which covers both models' maxima. Trajectory EPE uses the original 96 sites over frames 1–27.

Figure 3 uses the same rendering conventions as Appendix B.3. Material colors follow the recorded simulation partitions.

Interaction sequences across benchmark conditions. Figure 15 complements the configuration examples in Figure 3 with full interaction sequences. The examples cover topology, complex geometry, material layout, finger count, tool compliance and combined shifts. The camera and scale are fixed over time within each row. Material colors denote the same recorded regions described in Appendix B.4.

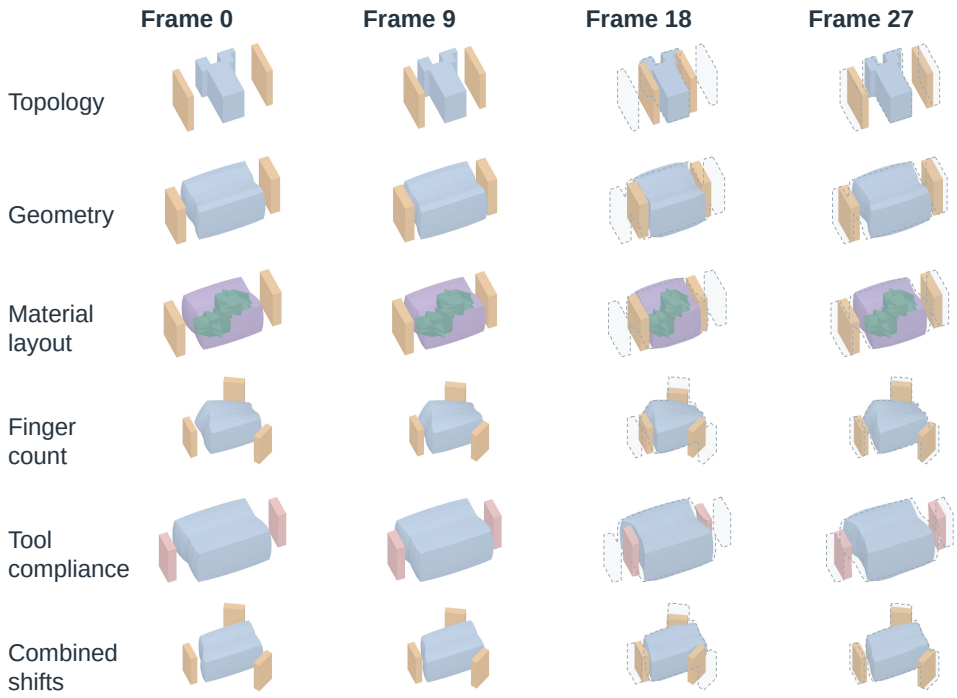

Figure 15: **Simulated grasping sequences across six benchmark conditions.** Columns show frames 0, 9, 18 and 27 without deformation scaling. Dashed translucent silhouettes mark frame 0. Blue denotes objects, peach rigid fingers and pink flexible fingers. In the material-layout row, mint and lavender denote 54 and 129.6 Pa regions; the transparent exterior reveals the internal softer region.

C GENERALIZATION, UNCERTAINTY AND EFFICIENCY

Details for Section 4.2 cover frozen confirmation, development results and the common inference workload.

C.1 FROZEN CONFIRMATION AND FULL COMPARISON

Confirmation evaluates frozen architectures, recipes, weights and metrics on 59 model–seed entries and 46 distinct V2 panels (2,714 model–panel predictions). The panel set contains 30 L1 and 16 L2 conditions: 44 test unseen identities, and two test new queries for seen identities. Two registry aliases are excluded. Development data were used for model design; confirmation data were accessed after freezing, with no subsequent model or threshold changes.

All 46 panels have EPE/dEPE and the 1,690 registered overlap queries at all three widths, passing the numerical criteria in Appendix B.2.2.

Supplemental frozen evaluations. Four additional ALPaCA entries complete seeds 42, 43 and 44 for both the MSE and NLL objectives. Seeds 43 and 44 use the unchanged 138-epoch recipe; all four weights were frozen before one additional evaluation on the same 46 panels. No tuning or seed selection followed access. The original 59 entries remain unchanged. A separate supplement evaluates three previously trained direct-attention seeds (Appendix E.2). These later evaluations are distinguished from the original simultaneous freeze.

ALPaCA-MSE EPE/dEPE is $1.1336 \pm 0.0108/1.0445 \pm 0.0113$ mm; ALPaCA-NLL gives $1.3828 \pm 0.0380/1.2302 \pm 0.0310$. The main table uses three-seed means for all prediction metrics. Overlap scores use the completed common-box evaluation of all 63 original and supplemental entries on the same 1,690 queries. Every query passes the numerical criteria at all three widths. At 2 mm,

Table 11: **Confirmation by benchmark level.** Three-seed means; errors in mm (↓), IoU/dIoU in % at 2 mm halfwidth (↑).

		Error (mm) ↓		Overlap (%) ↑	
Level	Method	EPE	dEPE	IoU	dIoU
L1	**DROP-Op**	**0.6109**	**0.5727**	**87.28**	**75.27**
L1	Pooled identification	0.7629	0.7234	84.29	70.51
L1	ICON	0.9473	0.8632	81.27	64.09
L2	**DROP-Op**	**1.0843**	**1.0024**	**79.34**	**65.85**
L2	Pooled identification	1.3249	1.2240	75.00	59.57
L2	ICON	1.6622	1.4294	70.99	52.90

ALPaCA-MSE IoU/dIoU are $79.97 \pm 0.06/64.91 \pm 0.24$; ALPaCA-NLL gives $76.57 \pm 0.34/61.05 \pm 0.44$ (percentages; mean $\pm$ training-seed SD).

C.1.1 Aggregate confirmation and uncertainty

The table below reports level-wise errors from the original frozen evaluation and overlaps from the common-box recomputation. DROP-Op and the pooled control share the full training recipe and differ in the organization of response statistics.

All three tube widths preserve the aggregate ordering among DROP-Op, pooled identification and ICON.

The fixed 5,000-draw identity-cluster bootstrap averages seeds first and retains the same identity across panels. Its intervals quantify individual paired comparisons. Against ICON, the EPE gain is 0.4204 mm [0.3918, 0.4513] and dEPE gain 0.3380 mm [0.3187, 0.3570]; against pooled evidence they are 0.1828 mm [0.1699, 0.1953] and 0.1753 mm [0.1633, 0.1866]. Bootstrap draws that leave any panel empty are rejected, preserving the same panel set in each accepted draw.

DROP-Op beats ICON on all 46 panels for both EPE and dEPE. Against ALPaCA-MSE, the counts are 44/46 and 43/46, and against DeformContact (global), 45/46 and 44/46. Comparing with the best baseline in each panel gives 38/46 and 33/46 wins. All comparisons use three-seed means, including the ALPaCA supplement. On partial actuation, DROP-Op's dEPE is 1.43055 mm versus DeformContact (global)'s 1.05410 mm.

Complete development results and all surface tolerances are in Appendix C.2; timing protocols are in Appendix C.3.

C.2 Development results and additional controls

DROP-Op uses training seeds 42, 43 and 44, each trained from scratch for 138 epochs. Its $288 \to 354 \to 96$ shared spatial blocks yield 326,688 parameters; width is chosen by parameter count. Seed 42 is the fixed parent for the single-parent mechanism and reuse studies. Across seeds, mean EPE/dEPE is 1.0571/0.9195 mm with standard deviations 0.0046/0.0097 mm.

IIDA uses 329,187 parameters and all 512 legal response vectors, with fresh 138-epoch training for three seeds. It averages EPE/dEPE 3.8785/3.3571 mm; 2 mm IoU/dIoU are $54.6321 \pm 0.7141/36.3944 \pm 1.0351$ (%). ALPaCA's 333,056-parameter NLL and MSE arms use matched initializations for seeds 42, 43 and 44 and the full 138-epoch recipe. Their different objectives and online posterior adaptation are specified in Appendix B.2.1.

The six ALPaCA entries share integration boxes and pass numerical checks at all three widths on 50 development panels; Table 12 uses the 48 common panels.

Table 12: **Development results at the primary 2 mm tube halfwidth.** Three-seed means on 48 panels. Bold marks the best score in each column. Larger tube widths are reported in Table 13.

	Error (mm) ↓		Overlap (%) ↑	
Method	EPE	dEPE	IoU	dIoU
DROP-Op	**1.0571**	**0.9195**	**83.09**	**70.34**
ICON	1.5095	1.2446	76.03	58.20
DeformContact (global)	1.5205	1.3088	75.86	59.41
DeformContact (local)	1.5763	1.3669	75.05	58.40
GNOT	1.7402	1.5028	73.52	54.52
GNO	1.7713	1.4483	73.63	55.30
Transolver	1.8390	1.5971	72.95	55.91
Transolver++	1.9650	1.7252	71.53	53.97
AdaptiGraph	2.0671	1.7453	73.89	58.50
EGNN	2.1717	1.7697	68.83	48.26
GeoTransolver	2.1926	1.8810	69.11	49.83
GINOT	2.2236	1.8921	69.04	50.68
MGN	2.3664	2.0704	65.57	44.97
NOMAD	2.6233	2.3167	63.86	44.55
RIGNO	3.0798	2.5824	55.60	37.43
UPT	3.1338	2.6183	55.62	35.49
Transformer	3.3227	3.0722	59.67	24.35
MLP	3.3916	3.1038	59.45	20.84
ALPaCA-MSE	1.4301	1.2404	78.34	62.99
ALPaCA-NLL	1.7077	1.4532	74.66	59.03

C.3 Unified efficiency and budget controls

A single session measures 30 implementations on two Fit objects, with query counts 1/4/16 and both cold and implemented reuse modes. The 360 workloads each use 15 timed samples after five warmups. All 60 numerical and cache checks pass. Measurements use an NVIDIA RTX PRO 5000 Blackwell with exclusive GPU compute and a shared host. Inputs are resident, so timings exclude sensing, disk access and host-to-device transfer. Each reported value averages the two object-specific medians. Reuse measurements reflect the implemented caches.

ALPaCA cost definition. The main table counts the complete mean-prediction path, including Cholesky solves, at 2.111321276 GFLOPs. A separate variance-producing audit costs 2.133688124/2.140239944 GFLOPs for 10/25 gripper sites; cached mean prediction costs 1.766610292/1.767917632 GFLOPs. Main timings use the mean-only path.

The expanded/original final-ten-epoch training-loss ratios are 0.470 (MLP), 1.440 (Transformer), 0.942 (RIGNO), 0.987 (UPT), 0.778 (GeoTransolver) and 1.006 (NOMAD). The Transformer's higher loss indicates an optimization limitation under the fixed recipe; the configuration below specifies widths and training.

C.3.1 Expanded-baseline configuration

All six controls retain their observation adapters, depth and training recipe. Only widths change, chosen using the preregistered 1.406898039 GFLOP target before accuracy evaluation. DROP-Op's compact cost is 1.337142519 GFLOPs. Selected widths are 16433 (MLP), 872 (Transformer), 201 (NOMAD), 232 (RIGNO), 160 (UPT) and 432 (GeoTransolver). These settings include 702,727,068 MLP parameters and 25,028,821 Transformer parameters; equal arithmetic does not imply equal capacity or latency.

Each expanded model trains from scratch for 138 epochs, 30,912 AdamW updates and effective batch eight under the original cosine schedule. The original response summaries remain, so this

Table 13: **Development overlap at larger tube halfwidths.** The models, seeds and panels match Table 12. All scores are percentages; higher is better.

Method	4 mm halfwidth		8 mm halfwidth	
	IoU (%)	dIoU (%)	IoU (%)	dIoU (%)
DROP-Op	**90.68**	**74.16**	**95.03**	**75.57**
ICON	86.07	62.53	92.45	64.57
DeformContact (global)	86.02	63.88	92.47	65.92
DeformContact (local)	85.49	62.84	92.18	64.90
GNOT	83.93	58.24	91.15	60.32
GNO	84.13	59.31	91.29	61.41
Transolver	83.52	59.77	90.79	61.71
Transolver++	82.50	57.77	90.22	59.84
AdaptiGraph	83.57	62.28	90.35	64.27
EGNN	81.65	53.58	89.98	56.03
GeoTransolver	80.64	53.03	89.11	55.00
GINOT	80.59	54.27	89.10	56.41
MGN	78.85	49.64	88.15	51.99
NOMAD	76.87	47.96	86.82	50.19
RIGNO	72.01	41.85	84.33	44.11
UPT	71.80	39.93	84.01	41.12
Transformer	73.43	25.33	84.39	25.98
MLP	72.72	20.28	83.67	20.41
ALPaCA-MSE	87.67	66.97	93.37	68.60
ALPaCA-NLL	84.86	62.78	91.66	64.53

Table 14: **Cold and reused prediction latency.** All entries are ms/query (↓), averaging two objects' medians. Reuse includes identification amortized over Q queries; inputs are resident. Bold marks the lowest latency in each column.

Method	Cold prediction 1 query	Reuse 4 queries	Reuse 16 queries
ALPaCA-MSE	10.262	5.430	**3.503**
ICON	38.098	21.633	15.840
MLP	**5.458**	**5.151**	3.695
Pooled identification	19.022	8.501	4.153
DROP-Op	19.535	8.827	4.187

tests scaling those implementations. Higher Transformer training loss limits conclusions about its optimized efficiency frontier. Table 15 reports all six results, including regressions.

Table 15: **Compute-matched expanded baselines.** One seed on 48 development panels. Parameters are in millions; GFLOPs cover cold prediction. Errors in mm; IoU/dIoU in % at 2 mm halfwidth.

	Cold prediction cost		Error (mm) ↓		Overlap (%) ↑	
Method	Params. (M)	GFLOPs	EPE	dEPE	IoU	dIoU
MLP	702.727	1.407	3.078	2.837	61.3	29.4
Transformer	25.029	1.396	3.838	3.516	56.9	21.2
NOMAD	0.759	1.406	2.661	2.359	63.9	44.6
RIGNO	1.582	1.399	3.083	2.483	56.6	38.0
UPT	0.910	1.351	3.065	2.545	57.7	37.1
GeoTransolver	6.965	1.403	2.198	1.853	69.9	50.4
DROP-Op	0.327	1.337	**1.059**	**0.931**	**83.1**	**70.3**

D Representation comparison protocol

This section supplies the controlled inputs and supervision for Section 4.3.

D.1 Representations with a shared frozen prediction framework

In Table 2A, each head is a small trainable network that maps its input representation to coefficients for the shared displacement decoder. Identification variants also fit coefficients to observed responses.

Each arm reuses the seed-42 DROP-Op geometry, population predictor and query features. Its 2,221-parameter head starts from a copy of the same trained parent head and is optimized for 138 additional epochs (30,912 AdamW updates) with the common data order and query samples. The full path therefore includes parent training plus head training. All heads produce $32 \times 12 \times 3$ coefficient slots for the shared displacement decoder; averaging can make those slots equal. Table 16 specifies the changed information pathway.

The inferred-scalar arm adds Fit material-label supervision and a 147-coefficient predictor, with geometry-group cross-fitting for training inputs. Fixed-basis controls bypass 11,084 learned-basis parameters. Shared features and the population predictor were jointly trained with the GPR representation.

Table 16: **Representation inputs and coefficient construction for Table 2A.** Geometry features and the displacement decoder are shared.

Representation	Information supplied and how it is used
Geometry features	Geometry features plus polynomial and sinusoidal encodings of normalized anchor coordinates; no response values.
Inferred material scalars	Geometry plus encodings of two inferred log moduli and an interface offset, broadcast across anchors.
Observed displacement history	Geometry plus the 16 observed 3D displacements at each anchor (48 values), directly mapped to coefficients.
Fixed response basis (pooled or spatial)	A rank-12 basis from known-command motion, velocity, variance and time-modulated motion replaces the learned basis in both support and query paths. Global coefficients average the local solutions.
Learned global response correction	Pool geometry and normal-equation statistics before the head; add a learned correction to the global ridge estimate.
Spatial identification followed by averaging	Compute the local identified coefficients, then average and broadcast them before querying.
Learned spatial response correction	Map local response statistics to a learned coefficient correction around the common global ridge estimate, without the final local solve.
GPR identified state	Use the learned response basis, anchor-specific equations and conditional regularized solve.

D.2 Complete predictors trained from scratch

For Table 2B, both field arms retain DROP-Op's geometry, action and query architecture but replace identification with material conditioning. Each has 327,155 parameters and learns all features and its decoder from scratch; DROP-Op has 326,688. Seeds 42, 43 and 44 follow the same 138-epoch recipe, 1,792 Fit queries and 30,912 AdamW updates. Material inputs replace support responses and use privileged simulation metadata.

For each tetrahedron e, the input contains its normalized rest centroid $\widetilde{c}_e$ and

$$h_e = \mathrm{MLP}\left[\widetilde{c}_e, \log(E_e/100\,\mathrm{Pa}), \nu, \log(V_e/\overline{V})\right]. \tag{26}$$

Here E_e is Young's modulus, ν is the recorded object Poisson ratio, and V_e is tetrahedron volume. Moduli come from stored element values or exact recorded material-region rules. All retained tetrahedra enter the encoder. The spatial arm pools their features at each anchor a with weights proportional to $V_e \exp(-\|\widetilde{x}_a - \widetilde{c}_e\|^2/0.5)$; the pooled arm uses volume weights shared by all anchors. A learned readout combines these features with anchor geometry to produce the same $32 \times 12 \times 3$ coefficient layout. Density and contact-property fields are outside this input set.

Evaluation reuses a diagnostic query selection fixed before inference. Two flexible-finger queries lack the required Poisson-ratio metadata and are excluded from every arm, leaving 62 common queries across 47 development panels. Queries average within panels, then panels equally, followed by the three-seed mean. Spatial-field EPE/dEPE is $2.2609 \pm 0.0647/2.0312 \pm 0.0928$ mm; pooled-field errors are $2.2717 \pm 0.0618/2.0360 \pm 0.0938$; GPR gives $0.9068 \pm 0.0121/0.8437 \pm 0.0127$. These comparisons evaluate response information against the specified elastic descriptors with joint feature learning.

E Response use and spatial evidence organization

Section 4.4 tests whether predictions use observed responses. Here we detail the response-swap tests and the attention and pooled-evidence controls for identification and spatial organization.

E.1 RESPONSE DIFFERENCES AND EXCHANGE CONTROLS

The 172 fixed training pairs match every legal input except response observations, with four queries per pair. For true fields D_a, D_b and predictions $\widehat{D}_a, \widehat{D}_b$, response-difference error is the mean point–time norm of $(\widehat{D}_b - \widehat{D}_a) - (D_b - D_a)$, in mm. A predictor that ignores responses has zero predicted difference. Exchanging evidence gives the partner prediction because all other inputs match; replay checks verify this equivalence.

DROP-Op's response-difference error is 0.2915 mm versus the zero-difference reference of 0.7931 mm. ALPaCA-MSE gives 0.3729 ± 0.0033 mm across three training seeds. Using seed 42 for both gives 0.2914 and 0.3736 mm. The 172 pairs form 87 exact-input groups; resampling these groups for the seed-42 comparison gives a descriptive ALPaCA-minus-GPR interval of [0.0719, 0.0923] mm. Geometric ancestry may correlate the groups. These are training-object diagnostics; Section 4.2 evaluates generalization.

IIDA and the earlier attention adapter each give 0.7931 mm. A fixed-weight audit verifies that response values reach both encoders, but their effects are attenuated by the learned decoder and initial-time subtraction. For query and initial decoder changes $\delta f_t, \delta f_0$, define

$$c = \frac{\|\delta f_t - \delta f_0\|_{\mathrm{av}}}{\|\delta f_t\|_{\mathrm{av}} + \|\delta f_0\|_{\mathrm{av}}},$$

where the norm averages point–time vector magnitudes. Final response effects are 0.000419 mm for IIDA and 0.000014 mm for the initial attention adapter, with mean ratios 0.1667 and 0.0179. Input/mask checks and 48 double-precision derivative checks pass. The direct-response attention control in Appendix E.2 addresses this weak response path.

E.2 A JOINTLY TRAINED CONTROL WITH DIRECT RESPONSE ATTENTION

This control keeps DROP-Op's geometry, action-memory and query-field backbone and replaces its local identifier with deterministic residual cross-attention. Each of 512 K4 tokens retains location, time and a learned action descriptor. The attention values are observed normalized displacements minus the shared prior, divided by support time. Identically zero initial-time tokens are excluded from attention normalization. A learned query-conditioned 3×3 map transforms the attended response, and multiplication by query time enforces zero initial displacement. Spatial, temporal and learned-descriptor distances determine attention weights. The model learns the attention and response maps jointly, with no closed-form coefficient solve.

The control has 326,927 parameters, versus DROP-Op's 326,688. Three seeds share matched backbone initializations and the fresh 138-epoch recipe in Appendix B.2.1. Evaluation uses the 48 distinct development panels and 172 matched-input training pairs.

The three-seed mean EPE/dEPE is 1.6048/1.4062 mm, with standard deviations 0.0148/0.0155 mm. Response-difference error is 0.4877 mm. Correct and exchanged response evidence give EPE 0.5577 and 0.9583 mm.

All three attention models distinguish responses, and exchanging context increases error. Appendix E.1 defines the response-difference metric.

Complete cold arithmetic is 1.440336 GFLOPs for ten gripper sites, versus 1.337143 for DROP-Op.

Post-confirmation evaluation. The three final attention checkpoints were frozen before one supplemental evaluation on the original 46 V2 confirmation panels (138 model–panel predictions). Architecture, recipe and seeds remained fixed; original GPR predictions use the same queries. This supplement follows the original confirmation freeze; V1 remains unaccessed.

Attention EPE/dEPE is 1.2930 ± 0.0021/1.1739 ± 0.0055 mm, compared with DROP-Op's 0.7756 ± 0.0015/0.7222 ± 0.0037. DROP-Op improves EPE on all 46 panels and dEPE on 45. The exception is heterogeneity-layout OOD, where attention has dEPE 1.2524 mm and DROP-Op 1.2937 mm. Table 3B reports this complement to the development response-sensitivity test.

E.3 SPATIAL EVIDENCE ORGANIZATION

Matched state capacity. To isolate spatial allocation at equal state capacity, the pooled-evidence control keeps 32 geometry-dependent prior centers C_j and precisions λ_j, the same 2,181-parameter architecture and initialization, the same geometry transfer and the same 32 solves. The two heads share initialization and are then trained separately. The only architectural change is

$$A_j^{\text{pooled}} = (\bar{G} + \lambda_j I)^{-1}(\bar{H} + \lambda_j C_j), \qquad (\bar{G}, \bar{H}) = \tfrac{1}{32}\sum_{j=1}^{32}(G_j, H_j), \tag{27}$$

instead of the local normal equations. Priors depend on geometry only in these controlled heads. Both states allocate $32 \times 12 \times 3 = 1,152$ coefficients, and both have nonzero spatial coefficient variation. Stored capacity is matched, while pooling changes the rank of the retained evidence.

Both heads share initialization, data order, query samples and optimization: 138 head epochs and 30,912 updates on one frozen parent, totaling 276 epochs including parent training.

With one head initialization and shared frozen features, pooled and local EPE/dEPE are 1.665/1.374 and 1.425/1.193 mm. Local identification improves the unrounded errors by 14.39%/13.17%, winning on 48/48 and 47/48 panels. Panels may repeat identities.

Joint training of local and pooled identification. The full pooled-evidence model is trained from scratch at 326,688 parameters, with exact seed-specific initial weights shared with the local model and the same 138-pass recipe. Gram, right-hand-side and squared-response statistics are pooled and then broadcast. Geometry, query readout, command history and learned blocks are otherwise unchanged. Development pooled-minus-local EPE/dEPE are 0.23838/0.220 mm, with 44/48 panel wins for each. The EPE identity-cluster interval is [0.19788, 0.28746] mm. A stratified identity-bootstrap sensitivity analysis retains panel membership rather than rejecting empty panels; both analyses report the full set of domains.

The spatial-error decomposition uses the exact identity

$$\frac{1}{N}\sum_i \|e_i\|^2 = \|\bar{e}\|^2 + \frac{1}{N}\sum_i \|e_i - \bar{e}\|^2.$$

Pooled-minus-main total, centroid and centered squared errors are 2.8388, 2.8601 and $-0.0213\,\text{mm}^2$. The EPE/dEPE improvement therefore coexists with slightly higher centered squared error. Gains also vary nonmonotonically with response heterogeneity: raw and panel-demeaned Spearman correlations are -0.3146 and -0.1659. The repeated identities induce correlations across panels. Spatial allocation improves the reported prediction errors without uniformly improving every spatial-error component.

E.4 VISUALIZING DOMAIN-WISE MECHANISM COMPARISONS

Figures 16 and 17 show how the mechanism comparisons vary across development conditions. Each panel averages the three training seeds after the query and identity aggregation in Section 4.1. The heatmap covers all 16 V2-L2 panels. Its signed reduction is $100(e_{\text{control}} - e_{\text{GPR}})/e_{\text{control}}$, so positive values favor DROP-Op. Improvements are substantial for joint shifts and finger-count changes, while pooling remains competitive or better on several material and contact conditions.

E.4.1 PAIRED ERRORS ACROSS THE FULL DEVELOPMENT EVALUATION

The scatter plots in Figure 17 retain all 48 distinct development panels. Against direct response attention, DROP-Op lowers EPE on 47 panels and dEPE on 46. Against pooled identification, it lowers both errors on 44 panels. The aggregate EPE/dEPE values are 1.0571/0.9195 mm for DROP-Op, 1.6048/1.4062 mm for attention and 1.2955/1.1392 mm for pooling. These development comparisons complement the frozen confirmation results in Table 3B.

E.5 SPATIAL PREDICTIONS UNDER ALTERNATIVE EVIDENCE MECHANISMS

Figure 18 visualizes the fully trained attention and pooled-identification controls alongside DROP-Op. It uses seed 42 and the first registered development query in two selected domains, topology

OOD condition	DROP-Op error (mm) ↓		Reduction vs. direct attention (%) ↑		Reduction vs. pooled identification (%) ↑	
	EPE	dEPE	EPE	dEPE	EPE	dEPE
Contact-regime change	1.898	1.762	+2.8	-2.0	-2.2	-2.6
Rate and loading history	1.352	1.212	+16.9	+13.9	+3.1	+6.1
Five-factor joint shift	0.982	0.955	+67.3	+68.0	+39.3	+38.6
Complex geometry	1.177	1.019	+36.8	+40.4	+12.0	+14.5
Geometry and material composition	0.568	0.538	+63.1	+63.9	+21.2	+21.8
Finger count	0.948	0.864	+67.4	+69.4	+37.7	+38.6
Material contrast and correlation	1.216	1.163	+7.4	+8.5	-2.5	-3.0
Material layout	1.291	1.242	+2.5	-10.3	-11.3	-15.0
Large deformation	1.373	1.274	+44.6	+46.3	-8.2	-4.4
Object and gripper composition	0.859	0.825	+44.7	+44.6	+12.5	+10.5
Partial actuation and asymmetry	1.766	1.517	+30.1	+0.2	+30.6	+29.6
Rare-tail joint distribution	0.745	0.709	+53.9	+50.0	+21.9	+21.4
Rigid gripper shape	2.055	1.990	+59.1	+59.1	+16.2	+15.2
Topology and load paths	0.862	0.816	+41.1	+40.1	+26.4	+25.1
Topology and boundary composition	0.601	0.578	+36.5	+30.6	+28.5	+27.5
Flexible fingers	0.785	0.708	+39.5	+42.3	+20.6	+22.4

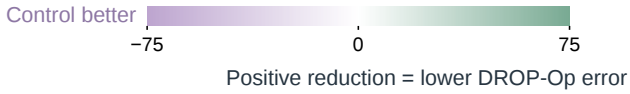

Figure 16: **Mechanism comparisons across V2-L2 development domains.** DROP-Op errors are in mm; signed percentages show reductions relative to each control. All entries average three training seeds. The diverging color scale is centered at zero, with positive values favoring DROP-Op.

and material layout. All three models retain their original 138-epoch training results. The topology example favors DROP-Op, with trajectory EPE of 0.646 mm versus 1.035 mm for attention and 0.980 mm for pooling. In the material-layout example, attention and pooling give 0.984 and 1.183 mm, versus DROP-Op's 1.505 mm. These spatial views complement the three-seed comparisons by locating both improvements and residual errors.

Surfaces use the common interpolation procedure in Appendix B.4. Each case uses the same camera and scale for all methods. Reference reconstruction RMSE against the dense simulated trajectory is 0.205 and 0.430 mm for the topology and material-layout cases, respectively. Curves and trajectory errors use the original 96 prediction sites.

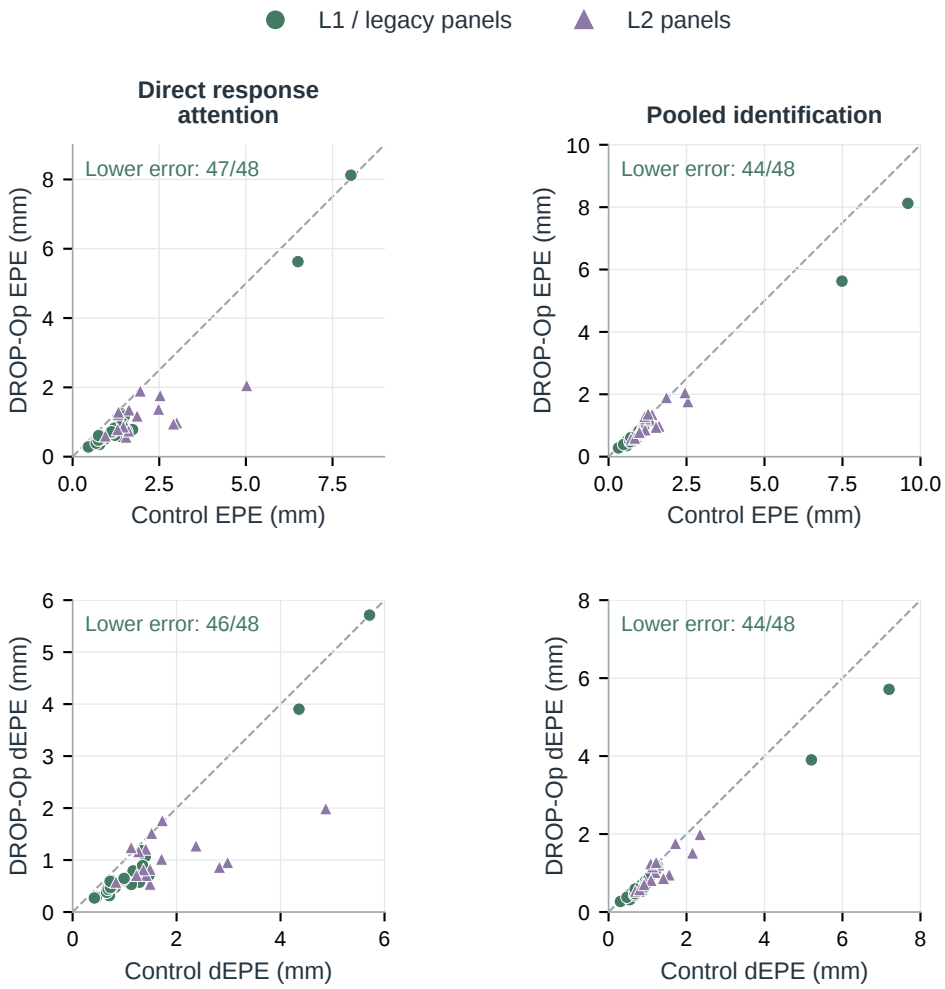

Figure 17: **Paired errors on all 48 development panels.** Each point is a three-seed mean. Circles denote 32 L1 and legacy panels; triangles denote 16 L2 panels. The diagonal marks equal error. Counts give the panels with lower DROP-Op error.

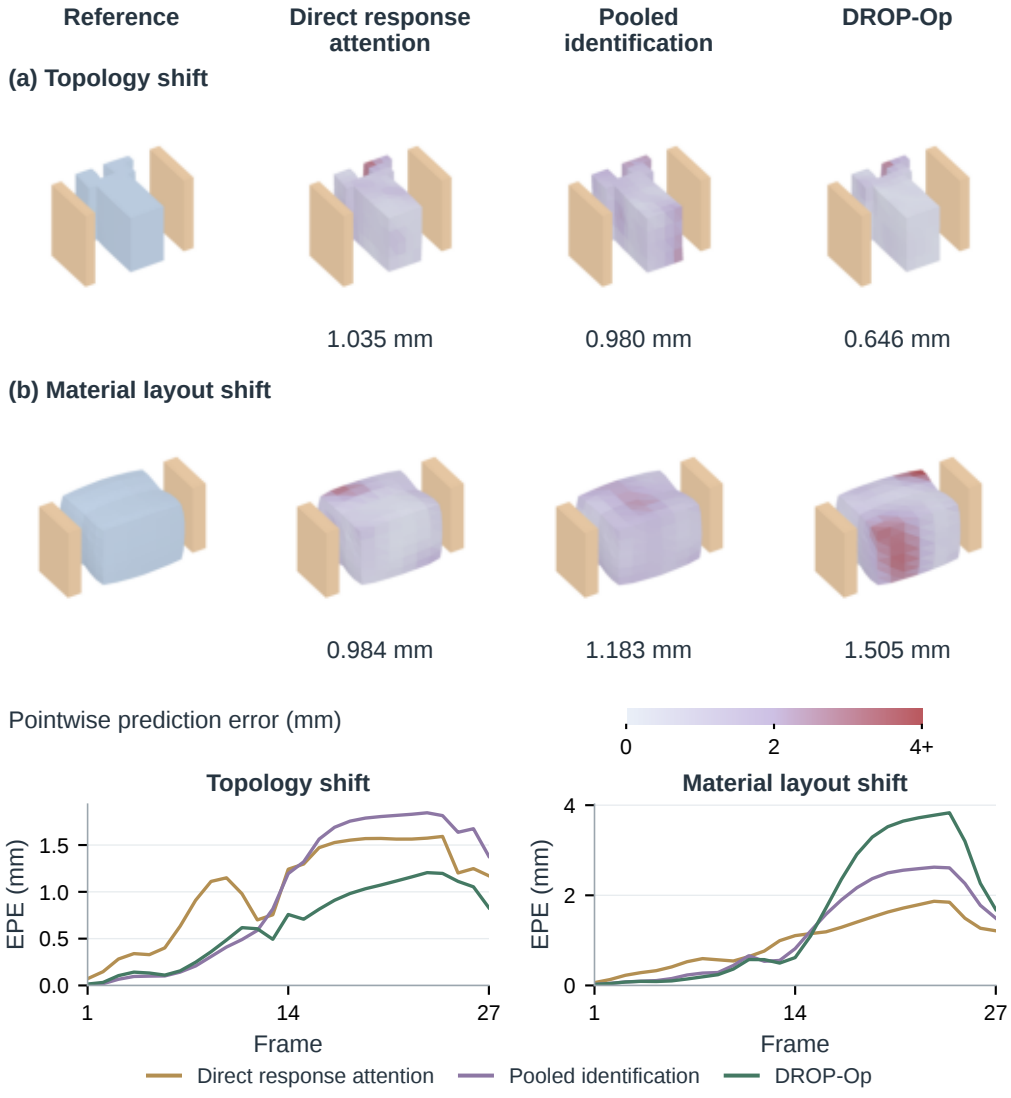

Figure 18: **Spatial prediction with different evidence mechanisms.** The meshes show frame 27; the curves show per-frame EPE and the labels show trajectory EPE. A common color scale shows interpolated prediction error, capped at 4 mm. Peach denotes the gripper.

F BACKBONE AND EXTERNAL-TASK ADAPTATION

This section supplements Section 4.5 with the transferred mechanism, source training, observations and query protocols, including conditions in which adaptation loses accuracy.

F.1 BACKBONE TRANSFER WITH MATCHED CONTROLS

Each backbone is a fixed seed-42 checkpoint trained for 138 epochs. Each adaptation uses three head initializations trained for 12 epochs (2,688 updates), in addition to parent training. Spatial and pooled identification have 2,874 trainable parameters with matched decoders and allocated state; learned conditioning has 5,366. These are adapter replications on fixed parents. Figure 6 reports all aggregate comparisons.

AdaptiGraph local EPE/dEPE are 1.57662±0.03604/1.36972±0.03464 mm, improving its parent on 46/48 and 42/48 panels, and pooled identification on 43/48 and 42/48. Its three-initialization mean beats the main model's three-training-seed mean on 9/48 panels per metric. Transolver local errors are 1.72519±0.00828/1.52070±0.00684 mm, improving its parent on 37/48 and 36/48, but beating the main model on none. Both error metrics improve in all 16 L2 panels for Transolver; easier L1 regressions are retained.

ALPaCA matched pooled/local errors are 1.45802/1.41760 mm EPE and 1.25981/1.21906 mm dEPE; local head SDs are 0.00485/0.00507 mm. Paired local advantages have 95% intervals [0.03783, 0.04331] and [0.03858, 0.04296] mm. AdaptiGraph/Transolver adaptations require four extra parent calls for support reconstruction. Their reported comparison covers EPE/dEPE; isolated timing and additional IoU are unavailable.

ALPaCA conditioning gives EPE/dEPE $1.45843 \pm 0.00051/1.26003 \pm 0.00042$ mm across these initializations.

F.2 NERD: SOURCE LEARNING AND FROZEN-STATE TRANSFER

The actuated Cartpole task predicts the cart and pole state over 100 steps under control commands. Gravity and pole mass/inertia changes test response-based adaptation to altered robot dynamics. NeRD + DROP-ID retains the native decoder f_{native} and adds a response-identified residual, $f_{\text{adapted}}(h) = f_{\text{native}}(h) + \phi(h)\widehat{W}$. The Transformer, native decoder and integrator remain fixed. The comparison uses learned, random and handcrafted bases, native prediction, and regression on native hidden features. All adapted methods receive the same four support trajectories.

Source training. Gravity and pole mass/inertia multipliers are the four corners of $\{0.75, 1.25\}^2$, with three generation seeds per corner. Each of the 12 episodes has four support and 64 query trajectories of 100 steps. Native 64-dimensional hidden features and normalized one-step residuals are cached. Normalization uses source support features only. A $64 \to 32 \to 8$ tanh network has 2,344 parameters; its basis is learned through the episode's ridge solve. With residuals R, basis $X = \phi_\theta(H)$, column RMS matrix S and $Z = XS^{-1}$,

$$W_\theta(\mathcal{C}) = S^{-1}(Z^\top Z/n + 0.1I)^{-1}Z^\top R/n, \tag{28}$$

$$\mathcal{L}(\theta) = \mathbb{E}_{\mathcal{C},q}\|\phi_\theta(h_q)W_\theta(\mathcal{C}) - r_q\|_2^2. \tag{29}$$

Each of three feature initializations uses 100 passes and 1,200 AdamW updates, with 256 sampled query transitions per episode/update. Learning rate decays from $3 \cdot 10^{-4}$ to 10^{-5}, weight decay is 0.01, and gradient clipping is one. Source-training cost is additional to the native checkpoint cost. Final features are frozen before target generation.

Target identification and prediction. Gravity/payload pairs are $(1, 1), (0.6, 1), (1.4, 1), (1, 1.4)$, with three fresh generation seeds each. Four random-action supports identify one state. It is reused for native-range random commands, sinusoidal commands of period 25 steps, and alternating 25-step blocks. Sine/block amplitudes are 0.75 of the native half-range. Each family has 16 queries with shared initial states. The metric averages wrapped-state L_2 error over all 100 rollout steps. Future query responses never enter identification.

Native-feature ridge fits 260 coefficients; learned, random and handcrafted bases fit 32. Feature and state hashes remain fixed across query families. Zero-response rollouts reproduce native predictions in all 12 condition/seed groups. Additional controls include mean correction and temporally mismatched evidence.

Results and variability. Table 4 gives equally weighted condition/action means. Learned features reduce error by 19.75% versus native NeRD, 7.21% versus generic ridge and 11.20% versus random features. Wins are 9/12, 6/12 and 10/12 groups, respectively. Feature-seed standard deviations are 0.026010 for learned, 0.078904 for random and 0.052758 for mismatched evidence. At payload 1.4, learned-basis random/sine errors exceed native; nominal block commands also slightly favor native. The handcrafted basis wins all 12 groups versus native and 10/12 versus learned features.

The 8,064 trajectory–model records use one parent checkpoint and correlated target conditions. The feature basis is trained on Cartpole source conditions and reused with newly identified states on target conditions.

F.3 DISCO: IDENTIFYING GENERATOR-COMPOSITION WEIGHTS

DISCO + DROP-ID adapts response identification to the released advection–diffusion operator (Serrano et al., 2026). The fixed checkpoint has 16,183,668 parameters. All methods observe 16 frames at 256 sites and forecast 34 frames for the first 128 public seed-124 trajectories. Advection/diffusion metadata are excluded from inputs. These exposed data were used during external-task exploration.

Composition replacement. The transferred DROP-Op procedure uses observed responses to identify a reusable coefficient state. Here the coefficients weight frozen dynamics generators, replacing native splitting-based composition. Standard RK4 integrates their weighted sum; generators f_j and the decoder remain fixed. We identify nonnegative weights for at most two generators:

$$\dot{u} = \sum_{j \in S} \widehat{w}_j f_j(u), \qquad |S| \le 2. \tag{30}$$

For the 15 observed transitions, set $A_{t,j} = \frac{h}{2}[f_j(u_t) + f_j(u_{t+1})]$, $y_t = u_{t+1} - u_t$, and $Z = AD^{-1}$, where D contains column norms. Each singleton/pair solves

$$\widehat{z}_S = \arg\min_{z \ge 0} \|Z_S z - y\|_2^2 + 10^{-3}\|z\|_2^2, \qquad \widehat{w}_S = D_S^{-1}\widehat{z}_S. \tag{31}$$

The smallest support objective selects S. The cached state stores its indices and coefficients; future frames enter neither selection nor fitting.

Controls and computation. The common eight-entry dictionary uses source mechanisms and seed 42. Native splitting uses beam width 2, maximum length 3 and its 5% stopping threshold. The unit-weight control searches the same singleton/pair subsets with the same RK4 predictor but fixes nonzero weights to one. Latent Adam uses 24 updates, 49 support forwards and 24 backwards. Gauss–Newton uses four rounds and 24 support evaluations. Neural weights remain fixed; adaptation budgets differ. Mismatched evidence shifts increments by seven steps, and zero updates recover direct predictions.

The encoder's random spatial permutation is shared across methods by restoring its random-number state. Source frame spacing is $10/99$ and evaluation step is $h = 0.1$. Independent double-precision replay differs by at most 1.304×10^{-7} in the main study. Queries from the identified state average 0.47846 s versus 0.47623 s for direct prediction. Charging shared feature extraction and fitting gives a 0.52358 s cold upper bound. Native search averages 1.96986 s on another GPU, so these timings describe their stated shared-GPU workloads rather than an isolated speed comparison.

Sensitivity to dictionary size. On the first 16 exposed trajectories, adding 24 seed-43 entries gives 32 entries. Native search uses beam width 3 and maximum length 5. Identification keeps its rule and regularization but enumerates 528 subsets instead of 36. Both the frozen eight-entry method and matched larger dictionary are evaluated. The larger native search outperforms the frozen eight-entry candidate, while enlarging the identifier's dictionary worsens its error. Dictionary choice and search budget therefore delimit the benefit. The original 256-entry search is not evaluated.

Table 17: **Dictionary-size sensitivity on 16 trajectories.** Entry counts specify the available generator dictionary.

Method	Dictionary entries	Relative L_2 error ↓
Native splitting	32	**0.042937**
Frozen DROP-ID	8	0.047569
Unit-weight composition	32	0.078180
DROP-ID	32	0.252255
DROP-ID with mismatched evidence	32	0.191568
DISCO direct prediction	—	0.309743

F.4 Other external tasks and limits of transfer

Table 18 reports additional interfaces using each task's native metric and horizon. The two reaction–diffusion studies use DISCO's Gray–Scott task (Serrano et al., 2026) and PDEBench's FitzHugh–Nagumo task with its released FNO (Takamoto et al., 2022; Li et al., 2021). CoDA tests adaptation to new Lotka–Volterra systems (Kirchmeyer et al., 2022); EqCollide tests continuation of deformable-body collisions (Chen et al., 2026). Readout replacement is tested in DeformContact contact deformation (Saleh et al., 2024), NeRD ANYmal dynamics (Xu et al., 2025), and AdaptiGraph granular manipulation (Zhang et al., 2024). Each study adapts the identification procedure within the external model, using task-specific observations and features.

The contact-pendulum replacement also increases 100-step error (0.780415 to 1.508709); restricting ANYmal adaptation to velocity outputs still leaves higher error (0.671171 vs. 0.627408). Numerical PDE updates do not consistently surpass native iteration or Anderson acceleration. These outcomes motivate checking observation coverage, module sensitivity and rollout horizon when applying identification to a new model.

Table 18: **Additional external-task results.** Comparisons are within each task. Positive relative error changes indicate degradation from the native reference.

External task	Evaluation	Adaptation result
DISCO: Gray–Scott	40 environments; disjoint query initial conditions	Relative L_2 0.056382: below native encoder 0.058230, above beam search 0.055029. The native-prior follow-up also remains behind beam search.
CoDA: Lotka–Volterra	Native new-system/new-initial-condition task	Local correction raises official relative error 28.83%; pooled correction raises it 11.861% despite lower MSE.
EqCollide: collisions	100 cases; 20 observed, 40 future frames	Initial pooled correction lowers MSE 0.2464%; revised spatial correction raises MSE 0.6548%. Sensitivity checks reject most cases.
FNO: FitzHugh–Nagumo	Nine test environments, four initial conditions; 91-step forecast	Native MSE 0.003168; spatial 0.355663, pooled 0.268214, attention 0.161957. All three adapter initializations lose on every case.
Native readout replacement	DeformContact, ANYmal and AdaptiGraph tasks	Full replacements worsen each native metric. An unshrunk DeformContact variant lowers point error but raises centered shape error.

G State reuse, readouts and action selection

This section supplements Sections 3.2 and 4.6 with readout definitions, candidate-action objectives and the composition interface.

G.1 Task-operator composition interface

A task-specific program can compose an identified state, a new command and a readout as $\widehat{\mathcal{O}}_j = \rho_j \circ \mathcal{D}_\theta \circ \mathcal{I}_\theta$. The reusable interface enforces the following contracts for manually written or AI-assisted programs (Liang et al., 2023; Li et al., 2026a).

Identification uses initial geometry and observed interactions. Reuse requires compatible geometry, interaction conditions and model versions. Each readout specifies its output frame and physical units. The program records these settings and which observations were used. Displacement readouts use metres.

G.2 Readout-error bounds

For $E = \widehat{D} - D$, centroid $c(D) = N^{-1} D^\top e_N$, centered RMS $s(D) = \|P_c D\|_F / \sqrt{N}$, and maximum displacement $m(D) = \max_i \|D_i\|_2$,

$$\|c(\widehat{D}) - c(D)\|_2 \le \|E\|_F / \sqrt{N}, \tag{32}$$

$$|s(\widehat{D}) - s(D)| \le \|P_c E\|_F / \sqrt{N}, \tag{33}$$

$$|m(\widehat{D}) - m(D)| \le \max_i \|E_i\|_2. \tag{34}$$

Table 19: **Candidate-action selection.** Regret is excess objective cost above the best candidate. Objective cost combines goal distance and deformation; it is not computational cost. Reductions compare DROP-Op with the listed best baseline across 364 identity–panel groups.

		Regret (mm) ↓		Reduction (%) ↑	
Deformation weight	Best baseline	Baseline	DROP-Op	Regret	Objective cost
0.0	DeformContact (global)	0.38713	**0.23933**	38.18	3.38
0.5	AdaptiGraph	0.45023	**0.24823**	44.87	3.28
1.0	AdaptiGraph	0.52936	**0.30813**	41.79	2.81

Cauchy–Schwarz proves the first inequality; the reverse triangle inequality proves the other two. These bounds connect field prediction to the three reported task readouts. If predicted candidate costs satisfy $|\widehat{J}(u) - J(u)| \leq \epsilon$ uniformly, minimization of $\widehat{J}$ gives $J(\widehat{u}) - \min_u J(u) \leq 2\epsilon$ by adding the two approximation errors.

G.3 Shared task readouts and candidate-action selection

Figure 7 applies the same three deterministic readouts to 18 model families, three training seeds and 48 development panels (70,524 model–query records). The cohort comprises DROP-Op and the 17 non-ALPaCA baselines in Table 12; the later ALPaCA additions are outside this readout study. No task head is trained.

Replay checks on 64 states and 256 commands per compact model reproduce uncached predictions exactly. Appendix C.3 measures end-to-end cost.

For decisions, all models see the same K4 evidence and initial geometry and choose among the same candidate commands. Six goals lie along signed coordinate axes at 5% of initial RMS radius. The objective adds centroid-to-goal distance and centered-deformation RMS weighted by 0, 0.5 or 1. The decision rule selects an action using predicted outcomes, and the recorded outcome is then used to score that choice. Uniform expected choice and the candidate oracle are controls. The study has 364 identity-panel groups, no exclusions and no new robot trials. Regret measures the excess realized objective relative to the best available candidate.

Bootstrap intervals condition on the fixed development panels and trained seeds.

H Observation coverage and application conditions

This section supports Section 5 with probe-coverage and changed-tool tests. Appendix F gives the external-task conditions and outcomes.

H.1 Effect of the number and choice of observed interactions

Figure 19 evaluates support coverage using the three fixed compact training seeds, 64 fixed queries and 48 panels. Model weights are fixed throughout the comparison. The same support set supplies K4, all six two-probe subsets (K2), and the no-response condition (K0), giving 1,536 records. Subsets retain the original probe identities and observation masks.

K4 has lower aggregate error than every K2 subset, although some individual panels favor fewer probes. Appendix H.2 tests interface changes.

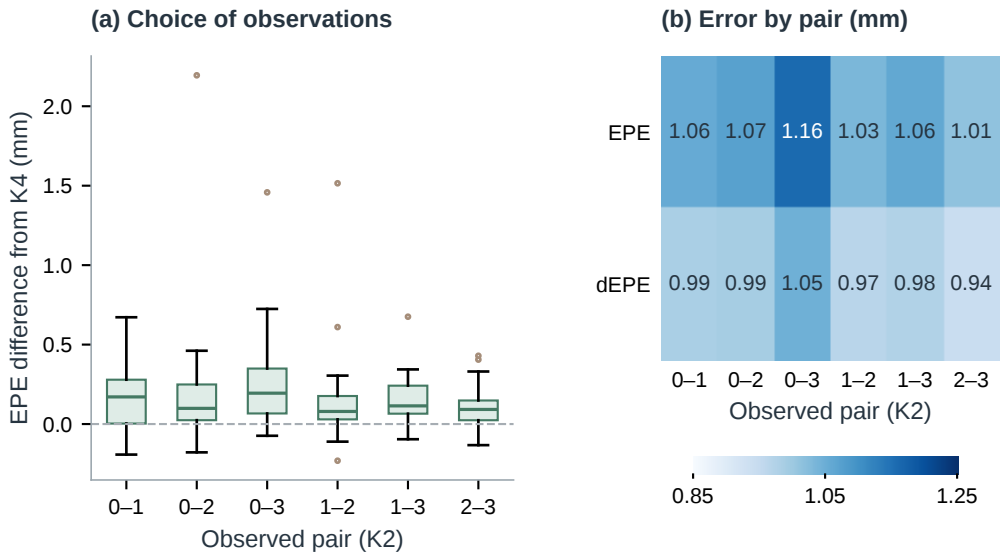

Figure 19: **Sensitivity to observed probes.** Left: panel-wise EPE differences from K4; positive values favor K4. Right: aggregate errors in mm. Pairs 0–1 through 2–3 select two of the four observed interactions. K4 uses all four; K0 uses none. Values average three seeds. Boxes show medians and interquartile ranges; whiskers reach the extreme observations within 1.5 interquartile ranges.

H.2 Re-identification after an interaction-interface change

New-tool evidence improves prediction, while mixing old- and new-tool responses performs worse than the same number of new-tool responses alone. Both directions of the single matched tool pair are evaluated.

Table 20: **Re-identification after changing tools.** One matched tool pair, both directions, eight queries. Errors in mm (↓); new-tool observations count toward acquisition.

Observed responses	EPE (mm) ↓	dEPE (mm) ↓
Four old-tool responses	3.9260	3.8229
One new-tool response	1.1011	0.9223
Two new-tool responses	0.6716	0.6232
Three old + one new response	2.2717	2.1929
Two old + two new responses	1.4340	1.3837
Four new-tool responses	**0.4902**	**0.4547**

H.3 MATHEMATICAL AND APPLICATION CONDITIONS

The response family is specified by its initial state, interface, commands and observed channels. Passive means externally excited; energetic passivity additionally requires a storage-function and supply-rate analysis (Willems, 1972). Parameter recovery requires observations that distinguish the parameters. The sufficiency result assumes fixed noiseless linear maps, and readout bounds propagate field error rather than calibrated uncertainty.

Positive regularization gives a unique conditional solve. With G_i, C_i, λ_i fixed, $\delta A_i = (G_i + \lambda_i I)^{-1}\delta H_i$ and $\|\delta A_i\|_F \leq \|\delta H_i\|_F/\lambda_i$. Changes in evidence also affect features and regularization, requiring the full differential or update in Appendix A. Reuse assumes compatible geometry, initial state and interface. Changed evidence requires renewed statistics and regularization; loading-history inference needs its own observations. New resolutions and task interfaces require corresponding feature and decoder validation.