G-INVERSE: GENERATIVE NEURAL INVERSE DYNAMICS FOR DEFORMABLE OBJECTS

Anonymous authors
Paper under double-blind review



Figure 1: **Generated deformable dynamics.** Illustration of our model generating object motions with distinct deformation behaviors across geometries and materials.

ABSTRACT

Predicting a deformable object's response to a new interaction requires information that its geometry alone does not provide. An observed motion reveals how the object responds to a particular loading, but fitting that motion does not establish transfer to another interaction. We present G-Inverse, which conditions a new trajectory on one complete observed 3D response with known loading and a separately specified query condition. Our approach separates a geometry-conditioned deformation representation from demonstration-conditioned dynamics. A temporal encoder summarizes the observed coordinates, velocities, and loading into object-specific response context. This context conditions latent acceleration prediction under the query loading. Integration and a frozen decoder produce local deformation, which is combined with global rigid motion to generate the world-space trajectory. We evaluate target-pose fitting separately from cross-condition rollout prediction. On held-out geometries, representation finetuning reduces mean fitting error on all four reported metrics. The dynamics evaluation compares temporal architectures using complete-query and end-point errors.

1 INTRODUCTION

Objects with similar shapes can respond very differently to the same interaction. A soft toy may recover after impact, a plastic object may retain permanent deformation, and a granular object may spread even when their initial geometries are comparable. Static appearance alone does not determine these behaviors. An observed motion sequence, however, reveals how an object deforms and recovers under a particular interaction. We ask how to use one complete observation to predict the same object's response to a different loading condition.

The difficulty is that a trajectory combines the object's intrinsic response and deformation history with its initial state, external forces, and contact. A model can fit the observed motion without learning information that remains useful under another interaction. Our target is therefore a conditional dynamics model: it uses a complete observed response and its known loading to predict a separate

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.

query trajectory. The query is specified by its initial state and physical condition; its future states are prediction targets rather than inputs.

We present G-Inverse, a framework for *generative neural inverse dynamics*. Here, inverse inference maps an observed force-conditioned response to object-specific dynamic context; forward prediction uses that context under a separately specified interaction. The context is trained for trajectory prediction without material labels as network inputs or supervision. It need not be a unique identification of the object's constitutive parameters.

Our central idea is to separate what the geometry can represent from how the object responds to loading. A finetuned, frozen geometry-conditioned decoder and an active subspace provide compact deformation coordinates for each object. Observations and predictions use this same coordinate system. A temporal encoder summarizes the demonstration coordinates, velocities, and known loading into response context. A conditioned dynamics model then predicts latent acceleration from the evolving query state and physical inputs. Integration advances local deformation and global rigid motion, while contact is evaluated on the predicted state. This formulation keeps the observed response fixed as context while the new interaction drives the query rollout.

The response encoder and query model are trained jointly across diverse geometries, four material families, and five interaction types. At inference, network weights and geometry-specific subspaces remain fixed. The complete demonstration and query initial state are fitted in the frozen deformation space, and the query is then predicted autoregressively. The reported query benchmark starts from the simulated rest state; it tests changed interactions rather than a sweep over arbitrary initial deformations or velocities.

We evaluate representation and dynamics separately. Target-pose fitting on held-out geometries tests the deformation representation; prediction against independently simulated query trajectories tests local deformation and global motion under changed conditions. Representation finetuning improves all four reported mean fitting metrics over released NeuROK (Table 1). The temporal-architecture results compare complete-query and end-point errors on a shared 79-query subset. The source corpus contains 9,100 trajectories from 4,520 geometries: 8,627 training trajectories from 4,281 geometries and 473 validation trajectories from 239 disjoint geometries.

Our contributions are threefold:

- We formulate prediction under a new interaction as generation conditioned jointly on one complete observed response and an explicit query condition.
- We develop a demonstration-conditioned acceleration model in a frozen, geometry-specific deformation space, combining local shape evolution with predicted-state contact and global motion.
- We evaluate the representation and query dynamics separately, reporting target-pose fitting, cross-condition prediction, and temporal-architecture comparisons on geometry-disjoint data.

2 Related Work

Inverse Physics and Material Identification. Inverse-physics methods identify material parameters or constitutive laws by matching simulated and observed motion, often through differentiable continuum simulation (Qiao et al., 2022; Li et al., 2023b; Cai et al., 2024; Ma et al., 2023; Mittal et al., 2025). Video-based system identification extends this idea beyond a single material class (Wang et al., 2015; Takahashi & Lin, 2019; Zhao et al., 2025; Liu et al., 2026; Rho et al., 2025). Neural rendering and simulation pipelines further recover motion, material, or physical laws from visual observations (Li et al., 2023a; Cao et al., 2024; Jiang et al., 2025; Lin et al., 2026; Kaneko, 2024; Ma et al., 2022; Liang et al., 2024b; Zheng et al., 2024; Le et al., 2025). We instead amortize inference across objects: one complete force-conditioned response produces context for a different interaction, without using material labels as network inputs or supervision.

Differentiable Physics and Deformable Manipulation. Differentiable simulators expose gradients through continuum, contact, cutting, and cloth dynamics (Hu et al., 2019; Geilinger et al., 2020;

Heiden et al., 2021; Qiao et al., 2021; Liang et al., 2019; Li et al., 2022), while physics-informed networks offer a complementary PDE formulation (Raissi et al., 2019). PlasticineLab, RoboCook, and RoboCraft optimize or learn long-horizon deformable manipulation (Huang et al., 2021; Shi et al., 2023; 2024), and Virtual Elastic Objects learns reusable interaction models (Chen et al., 2022a). Our MPM corpus builds on continuum simulation for solids, snow, and rigid-body coupling (Sulsky et al., 1995; Jiang et al., 2016; Stomakhin et al., 2013; Hu et al., 2018), but the predictor itself is learned rather than differentiated through the simulator.

Learned Dynamics for Deformable Objects. Particle- and mesh-based graph networks learn local state transitions from action–state trajectories (Li et al., 2018; Sanchez-Gonzalez et al., 2020; Pfaff et al., 2020); latent models provide compact rollouts for deformable manipulation (Li et al., 2024a), and interaction-driven representations encode physical properties without explicit labels (Xu et al., 2019). These models amortize forward simulation. Our task additionally separates a complete demonstration, which determines object-specific context, from a distinct query state and loading that drive the forecast.

Neural Coordinates and Dynamic Scenes. Reduced-order models use compact generalized coordinates for high-dimensional deformation (Chen et al., 2022b; Wang et al., 2024); NeuROK amortizes a geometry-conditioned kinematic space across object categories (Geng et al., 2026), and learned implicit surfaces provide continuous geometry from point samples (Erler et al., 2020; Ma et al., 2020). Dynamic radiance fields use canonical warps, higher-dimensional embeddings, or space–time fields (Pumarola et al., 2020; Park et al., 2021a;b; Xian et al., 2020; Wang et al., 2023a); image-based and factorized variants improve fidelity and efficiency (Attal et al., 2023; Li et al., 2023c; Yu et al., 2023b). Dynamic Gaussian methods track persistent primitives or deformation fields (Yang et al., 2024a; Wu et al., 2024; Yang et al., 2024b; Luiten et al., 2024; Stearns et al., 2024; Kratimenos et al., 2024; Yu et al., 2023a). These representations reconstruct observed motion, whereas our state must predict a response to a separately specified force or contact.

Generative and Physics-Guided 4D. Static 3D generation developed through score-distilled NeRFs, variational objectives, novel-view synthesis, and Gaussian splatting (Poole et al., 2022; Lin et al., 2023; Wang et al., 2023b; Liu et al., 2023; Tang et al., 2024; Kerbl et al., 2023). Generative 4D systems add diffusion, video consistency, dynamic Gaussians, trajectory control, multi-view conditioning, mesh animation, and compositional scenes (Ling et al., 2024; Jiang et al., 2024; Bahmani et al., 2024b; Ren et al., 2023; Yin et al., 2023; Zeng et al., 2024; Bahmani et al., 2024a; Wu et al., 2025a; Liang et al., 2024a; Yenphraphai et al., 2026; Sabathier et al., 2026; Xie et al., 2025; Liang et al., 2026; Feng et al., 2025); perceptual and physical suites assess their outputs (Huang et al., 2024; Duan et al., 2025). Physics-guided variants couple neural 3D assets to continuum, spring–mass, or constitutive simulators (Xie et al., 2024; Feng et al., 2024; Zhong et al., 2024; Shao et al., 2024; Lin et al., 2025), distill video priors (Zhang et al., 2024; Liu et al., 2024a; Huang et al., 2025), or use image, language, and video controls (Li et al., 2024b; Liu et al., 2024b; Qiu et al., 2024; Fu et al., 2024; Wang et al., 2026; Wu et al., 2025b). These methods generate plausible motion from a static asset and condition; we observe one response and evaluate transfer to an independently simulated query.

3 Problem Formulation and Overview

Let a demonstration be a complete observed world-space trajectory together with its known loading, $\mathcal{D} = \{(\mathbf{x}_t^d, \mathcal{B}_t^d)\}_{t=0}^{T_d}$. A query specifies the same reference geometry, its initial local and global state s_0^q, and a physical condition $\mathcal{B}^q$. We seek a predictor of the complete query response,

$$\{\widehat{\mathbf{x}}_t^q\}_{t=1}^{T_q} = \mathcal{G}_\theta(\mathcal{M}_0, s_0^q, \mathcal{B}^q; \mathcal{E}_\phi(\mathcal{D})) \,. \tag{1}$$

Demonstration and query are paired by common geometry and constitutive parameters. The reference geometry is an input, whereas the constitutive parameters define the pair but are not provided as material labels. In the general formulation the frame-zero state and loading may differ. The present benchmark initializes the query from its simulated frame-zero rest state and changes the interaction. The predictor evaluates known fields and boundaries causally on each predicted state.

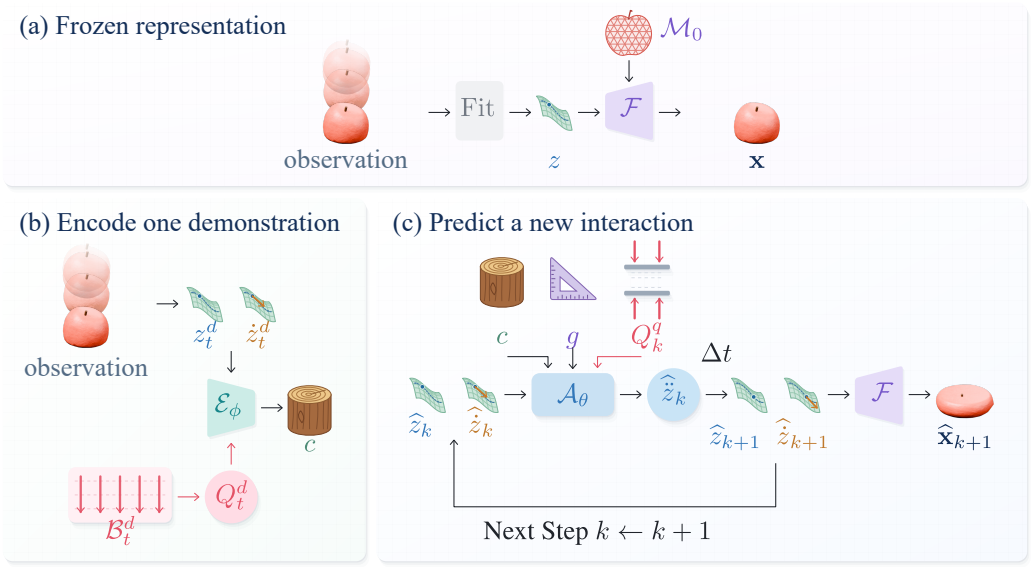

Figure 2: Overview of our method. (a) We map object-local deformation into a compact coordinate system using a frozen geometry-conditioned representation and a geometry-specific active subspace. Fit denotes posterior initialization and coordinate optimization. (b) A causal temporal encoder summarizes a complete demonstration and its applied loading into a response code c. (c) Given this code, the object geometry, current state, and causally evaluated physical inputs, a GRU or sliding-window causal Transformer rolls the latent state forward and the frozen decoder recovers the resulting shape. In this example, the demonstration is a drop under gravity and the query interaction compresses the object between two plates.

Our model factors this task into three stages. First, a frozen geometry-conditioned decoder maps world-space observations to compact object-local deformation coordinates while global rigid motion is represented separately. Second, a causal encoder summarizes the complete demonstration coordinates, velocities, and loading into an object-specific response context c. Third, a GRU or sliding-window causal Transformer query model predicts latent acceleration from c, the evolving query state, and causal physical inputs. Its learned force split uses a bias-free, geometry-gated external branch and AdaLN-Zero modulation in the internal branch. Global pose is advanced by causal force and torque integration, after which local deformation and rigid motion are composed in world coordinates. Appendix A.1 defines the variables and coordinate conventions used by each stage.

4 Compact Deformation Representation

We first construct a compact state space for object deformation. Its output is a set of reduced coordinates and a frozen decoder that maps those coordinates to deformed points. We finetune the representation on a broad deformation corpus, construct a geometry-specific active subspace, and fit each MPM trajectory in that subspace. Section 5 uses the resulting states to encode a complete demonstration and predict a query motion.

4.1 Representation Formulation

Represent the reference object by a fixed-connectivity mesh $\mathcal{M}_0$. Let $\mathbf{x}_0 \in \mathbb{R}^{3n}$ concatenate the positions of n corresponding reference points, and let $\mathbf{x}_t \in \mathbb{R}^{3n}$ denote their deformed positions at frame t. Both are expressed in normalized object-local coordinates, using one reference-shape normalization shared across materials and frames. Let $R_t \in \mathbb{R}^{3\times3}$ be the global rotation and $\mathbf{r}_t \in \mathbb{R}^3$ the translation. The world position of point i is

$$\mathbf{x}_{t,i}^{\text{world}} = R_t\big(\ell\mathbf{x}_{t,i} + \mathbf{o}\big) + \mathbf{r}_t, \tag{2}$$

where the scale ℓ and center $\mathbf{o}$ undo the reference normalization. This factorization assumes topology-preserving motion and a stable correspondence-based separation between rigid motion and local deformation.

Let $z \in \mathbb{R}^{20}$ be reduced deformation coordinates. The effective decoder $\mathcal{F}(\cdot; \mathcal{M}_0) : \mathbb{R}^{20} \to \mathbb{R}^{3n}$ maps z to deformed point positions. We omit the reference mesh from the notation when it is fixed. The decoder is evaluated at the geometry-conditioned code plus the active-subspace displacement. Its output is centered and aligned to the reference by the best-fitting proper rotation, fixing the rigid gauge while preserving the represented nonrigid shape. The same frozen, gauge-fixed map is used for coordinate fitting and dynamics prediction.

4.2 Geometry-Conditioned Representation Finetuning

We finetune the released NeuROK representation (Geng et al., 2026) on corresponding deformation sequences from DynamicObjaverse, DyMesh, and MPM. Its geometry-conditioned prior, deformation encoder, and dual-quaternion decoder are trained with reconstruction and KL objectives. We then freeze the complete representation so that its coordinates remain fixed during basis construction and dynamics learning. Appendix B gives the data normalization, architecture, loss, and optimization details.

4.3 Compact Coordinates for Simulation

For each geometry, we select 20 orthonormal nonrigid directions $B \in \mathbb{R}^{131072\times 20}$ and express the full code as $\mu_{\mathrm{cond}} + Bz$. The basis retains decoder directions with large point-space effects after removing infinitesimal translation and rotation; Appendix A gives its construction.

We first fit the unrestricted latent, project it into the geometry-specific basis, and refine only the rank-20 coordinates. The decoder Jacobian

$$J(z) = \frac{\partial \mathcal{F}(z)}{\partial z} \in \mathbb{R}^{3n\times 20} \tag{3}$$

maps latent velocity to point velocity, $\dot{\mathbf{x}} = J(z)\dot{z}$. After position fitting, we recover $\dot{z}_t$ causally by matching the surface secant velocity with the current decoder Jacobian. These fitted positions and velocities supervise dynamics training.

Appendix A derives the coordinate and force mappings, and Appendix D.1 discusses the corresponding fitting diagnostics.

5 Demonstration-Conditioned Dynamics

Given the deformation coordinates from Section 4, our dynamics model encodes one complete demonstration into a fixed feature vector and predicts a complete query trajectory. The representation and active subspace are frozen. During the query rollout, physical quantities are obtained from the predicted state and the declared query condition.

5.1 Encoding an Observed Response

Let $\mathbf{f}_{\mathrm{nr}} \in \mathbb{R}^{3n}$ denote the nonrigid component of point forces in the decoder's local coordinates. External fields are sampled at the current world-space points, and floor or plate contact is recomputed from the current predicted surface and velocity. After removing the instantaneous rigid force and torque components, we project the remaining force to deformation coordinates using

$$Q = J(z)^{\mathsf{T}}\mathbf{f}_{\mathrm{nr}} \in \mathbb{R}^{20}. \tag{4}$$

The same point masses, local frame, and reference scale are used during observation and rollout. Separate features retain the total world-space force and torque needed for global motion.

The response encoder reads the complete fitted demonstration sequence after feature scaling:

$$c = \mathcal{E}_\phi\left(\{z_t^d, \dot{z}_t^d, Q_t^d\}_{t=0}^{T_d}\right), \qquad c \in \mathbb{R}^{32}. \tag{5}$$

The GRU or causally masked Transformer encoder reads the complete demonstration before prediction; the resulting context remains fixed throughout the query rollout.

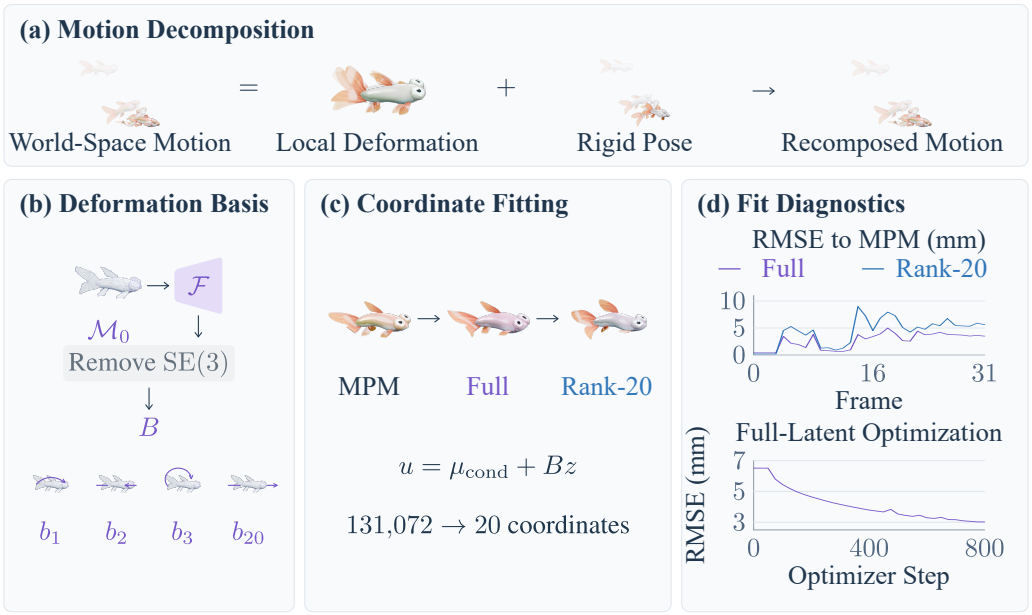

Figure 3: Compact coordinates on a held-out fish. (a) Rigid motion is separated from local deformation. (b) Decoder responses define a rank-20 nonrigid subspace. (c) A full-latent fit is projected and refined in this subspace. (d) Across 32 frames, the full and rank-20 fits obtain 3.02 mm and 4.96 mm RMSE to MPM, respectively.

5.2 Conditioned Acceleration Rollout

At integration microstep k, the dynamics model receives geometry features g, the fixed response feature c, generalized loading Q_k^q, predicted coordinates and velocity, the current rigid state, and net force and torque:

$$(h_{k+1}, \hat{\ddot{z}}_k) = \mathcal{A}_\theta(h_k, \hat{z}_k, \hat{\dot{z}}_k, Q_k^q, s_k^{q,\text{rigid}}, g, c). \tag{6}$$

We implement $\mathcal{A}_\theta$ with either a GRU or a sliding-window causal Transformer. A bias-free external-force branch is gated by geometry, while an internal-response branch is modulated by state, history, geometry, and c through AdaLN-Zero. Appendix B gives the architectural dimensions and feature scaling.

Here k is a discrete integration-step index, while τ_k is physical time. We advance the state with semi-implicit Euler integration:

$$\hat{\dot{z}}_{k+1} = \hat{\dot{z}}_k + \Delta t\, \hat{\ddot{z}}_k, \qquad \hat{z}_{k+1} = \hat{z}_k + \Delta t\, \hat{\dot{z}}_{k+1}. \tag{7}$$

Physical time advances as $\tau_{k+1} = \tau_k + \Delta t$, where Δt is the duration of one integration microstep. Each observation interval uses four microsteps. The equations use physical coordinates; normalization is applied only at network interfaces.

5.3 Global Motion and Contact

At each rollout microstep, the frozen decoder recovers the current predicted surface and its velocity. Known external fields and contact boundaries are evaluated on this surface, and their generalized contribution is recomputed before the network predicts latent acceleration. Net force and torque advance translation, angular momentum, and rotation with a causal rigid-body update. We combine the resulting pose with local deformation using Equation 2. Appendix A gives the force projection, rigid-component removal, and numerical details.

5.4 Training Objective

The response encoder and query model are trained jointly. Cross-condition pairs share geometry and material while changing the interaction; same-trajectory pairs provide auxiliary reconstruction supervision. Starting from the query frame-zero state, predictions are fed back autoregressively. We optimize

$$\mathcal{L}_{\mathrm{dyn}} = \mathcal{L}_z + \mathcal{L}_{\mathrm{local}} + \mathcal{L}_{\mathrm{world}} + \mathcal{L}_{\mathrm{trans}} + \mathcal{L}_{\mathrm{rot}}, \tag{8}$$

where the first term measures standardized rank-20 coordinates; the next two measure decoded local and world-space points normalized by object scale; and the final terms measure translation and rotation. At inference, the demonstration is encoded once and the model rolls out the query from frame zero. Feature normalization, rollout curriculum, and the full inference procedure are given in Appendix B.

6 Experiments

6.1 Experimental Settings

Datasets. Representation finetuning combines DynamicObjaverse, DyMesh, and generated MPM sequences, with 11,137 training clips and 619 validation clips from disjoint geometries. The dynamics corpus contains 9,100 trajectories from 4,520 geometries, spanning elastic, plastic, sand, and snow under drop, press, jet-pair, shear, and uniform-gust interactions. Representation evaluation uses 50 held-out clips. The reported dynamics evaluation uses 79 geometry-disjoint query cases, each paired with a complete demonstration of the same geometry and material under a different loading condition. Queries start from the simulated rest state. Appendix C gives the full splits and preprocessing; Appendix C.3 specifies the scoring protocol.

Baselines. For representation fitting, we compare with released NeuROK (Geng et al., 2026), a NeuralDeformationGraphs reproduction (Božič et al., 2021), and a KeypointDeformer baseline (Jakab et al., 2021). NDG receives five additional same-object support poses. For query prediction, we include static prediction, initial-velocity extrapolation, demonstration replay, and adapted GNS, NeuROK, and AnimateAnyMesh (Wu et al., 2025b) models. Their available inputs differ: AnimateAnyMesh receives an initial mesh and an interaction-class text prompt, while the projected-force NeuROK baseline uses the query initial state and condition. Appendix C.3 details all baseline inputs and fitting budgets.

Metrics. Representation evaluation reports NR-RMSE, Chamfer-L1, Chamfer-L2, and P90 after removing a best-fitting proper rigid transform. Dynamics evaluation reports world-space RMSE over the full predicted query and End-Point RMSE at its final frame, both in millimeters. Representation fitting and query prediction are evaluated separately. Dynamics errors aggregate per-trajectory mean squared errors with equal case weights before taking the square root. Method-specific point sampling is detailed in Appendix C.3.

Implementation Details. We finetune the released NeuROK representation with AdamW at learning rate 8×10^{-5} and batch size 16 per GPU, then freeze it and construct a rank-20 active subspace for each geometry. The response encoder produces a 32-dimensional context. Each temporal-architecture variant is trained independently with Adam at learning rate 10^{-3}, global batch size 128, and gradient clipping at 1. The rollout horizon grows from 4 to 8 to 16 frames and then to the full query. Semi-implicit integration uses four microsteps per observation interval. At inference, network weights remain fixed while demonstration and initial-state coordinates are fitted in the frozen representation. Appendix B gives the architecture, loss, normalization, and optimization details.

6.2 Deformation Representation

We first ask whether a frozen representation can express held-out object deformation before introducing dynamics error. We freeze a geometry-disjoint validation set and choose one maximum-temporal- separation input–target pair per geometry. Every method fits the same 512 corresponding

Table 1: Target-pose fitting on held-out geometries (mean; lower is better). Shape errors remove one proper rigid transform. Exact metrics, fitting schedules, and support inputs are given in Appendix C.3; standard deviations are reported in Table 5.

Method	Extra Support Poses	NR-RMSE	Chamfer-L1	Chamfer-L2	P90
NDG reproduction (Božič et al., 2021)	5	–	.0328	.0472	.0758
KeypointDeformer (Jakab et al., 2021)	0	.0297	.0243	.0301	.0820
NeuROK (Geng et al., 2026)	0	.0122	.0135	.0155	.0313
G-Inverse (Ours)	0	**.0115**	**.0124**	**.0142**	**.0296**

Table 2: Cross-condition query prediction. World and end-point RMSE are measured in millimeters. All rows use the same 79-query subset; method-specific vertex sampling and baseline inputs are detailed in Appendix C.3.

Method	World RMSE (mm)	End-Point RMSE (mm)
Static query frame	360.3	435.7
Query-velocity extrapolation	360.3	435.7
Demonstration replay	545.1	658.9
GNS†	270.6	509.0
NeuROK† (Geng et al., 2026)	324.0	411.6
AnimateAnyMesh† (Wu et al., 2025b)	369.4	453.8
G-Inverse (Ours)	120.5	172.7

† Adapted to the common rollout protocol.

points with its declared method-native optimization schedule. Because our representation models local deformation rather than global pose, all reported shape metrics remove one best-fitting proper rigid transform. Our finetuned model lowers mean error relative to released NeuROK on all four metrics in Table 1. It also lowers error relative to KeypointDeformer and the NDG reproduction on the metrics reported for each baseline. NDG receives five additional same-object support poses, which are disclosed in the table. Figure 3 separately illustrates unrestricted and rank-20 coordinate fitting on one held-out trajectory; Appendix D.1 provides additional interpretation.

6.3 Fixed-Demonstration Prediction under Changed Conditions

The primary dynamics evaluation encodes one complete demonstration and predicts an independently simulated query under a different interaction, using fixed geometry-disjoint demonstration–query pairs. Figure 5 shows a changed-condition comparison, while Figure 4 visualizes trajectory reconstruction.

Table 3: Temporal-architecture comparison on the same 79-query subset. All variants share the deformation representation, response conditioning, AdaLN-Zero, training data, and rollout objective. Values describe the evaluated checkpoints; Appendix C.3 specifies the reporting scope.

Demo Encoder	Query Predictor	Modulation	World RMSE (mm)	End-Point RMSE (mm)
GRU	GRU	AdaLN-Zero	149.6	228.1
Causal Transformer	GRU	AdaLN-Zero	120.5	172.7
Causal Transformer	Causal Transformer	AdaLN-Zero	134.3	195.4

On the 79-query subset, the causal demonstration encoder with a GRU query predictor has 19.4% lower world RMSE and 24.3% lower end-point RMSE than GRU–GRU (Table 3). The sliding-window query predictor does not improve on this configuration on this subset.

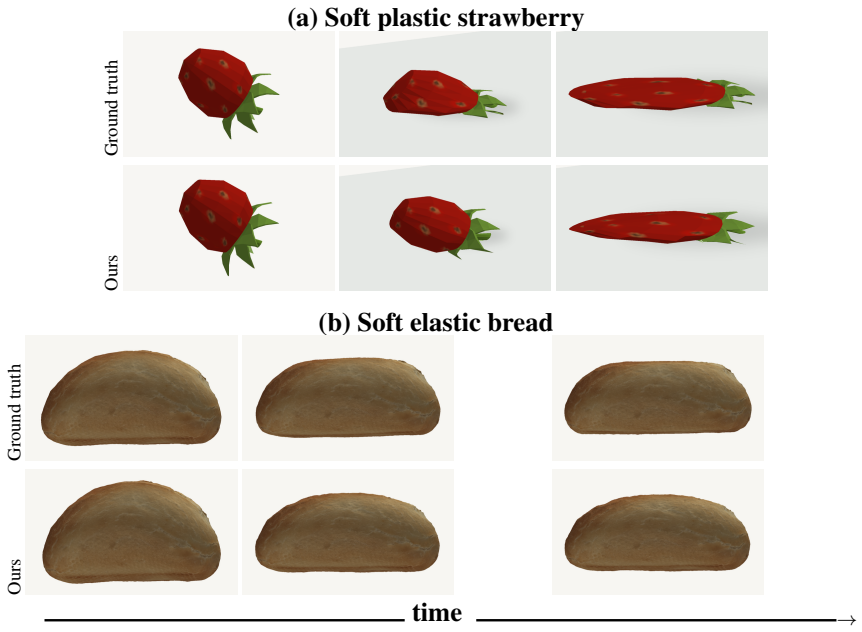

Figure 4: Trajectory reconstruction at three timestamps. (a) A soft plastic strawberry falls and retains a flattened shape. (b) A soft elastic bread loaf is compressed between symmetric plates.



Figure 5: Apple material interpolation under the same query external-force and contact condition at three matched timestamps. Rows show the simulated ground truth, the projected-force NeuROK baseline, and G-Inverse. All results use the original asset texture.

7 Conclusion

We presented G-Inverse, which uses one complete observed response and its known loading to condition deformation prediction under a separately specified interaction. Its frozen deformation space connects response encoding to acceleration prediction, local deformation, and global motion. Representation finetuning lowers held-out target-pose fitting error on all four reported metrics. The query experiments evaluate cross-condition rollout and compare temporal architectures, separately from representation fitting. All three temporal variants are compared on the same 79-query subset.

Limitation Discussion. The method assumes topology-preserving deformation, persistent correspondence, and a stable separation of rigid motion from local deformation. Reduced floor and plate contact laws can allow penetration during long rollouts. A single demonstration may be ambiguous when materials respond similarly to its loading. Predictions outside observed material, contact, or force regimes remain extrapolative, particularly for stronger or qualitatively different query interactions.

Ethics Statement

This work studies deformation prediction using simulated trajectories and existing 3D animation assets. Potential applications include animation, simulation, and deformable-object manipulation. Predicted trajectories may be inaccurate under unfamiliar materials, loading, or contact conditions; deployment in physical systems would require application-specific validation and safeguards. Synthetic outputs should be identified as model predictions rather than physical measurements. Use and redistribution of third-party assets and datasets remain subject to their original licenses and attribution requirements.

AI Use Statement

Generative AI tools assisted with code editing, manuscript drafting and language editing, preliminary figure sketches, and LaTeX figure and table formatting. These tools supported the research and writing workflow; their outputs are not independent evidence for scientific claims or experimental results. The authors are responsible for verifying the code, references, analyses, and reported results, and take responsibility for the final manuscript.

References

Benjamin Attal, Jia-Bin Huang, Christian Richardt, Michael Zollhoefer, Johannes Kopf, Matthew O'Toole, and Changil Kim. Hyperreel: High-fidelity 6-dof video with ray-conditioned sampling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16610–16620, 2023.

Sherwin Bahmani, Xian Liu, Wang Yifan, Ivan Skorokhodov, Victor Rong, Ziwei Liu, Xihui Liu, Jeong Joon Park, Sergey Tulyakov, Gordon Wetzstein, et al. Tc4d: Trajectory-conditioned text-to-4d generation. In *European Conference on Computer Vision*, pp. 53–72. Springer, 2024a.

Sherwin Bahmani, Ivan Skorokhodov, Victor Rong, Gordon Wetzstein, Leonidas Guibas, Peter Wonka, Sergey Tulyakov, Jeong Joon Park, Andrea Tagliasacchi, and David B Lindell. 4d-fy: Text-to-4d generation using hybrid score distillation sampling. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7996–8006. IEEE, 2024b.

Aljaž Božič, Pablo Palafox, Michael Zollhöfer, Justus Thies, Angela Dai, and Matthias Nießner. Neural deformation graphs for globally-consistent non-rigid reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1450–1459, 2021.

Junhao Cai, Yuji Yang, Weihao Yuan, Yisheng He, Zilong Dong, Liefeng Bo, Hui Cheng, and Qifeng Chen. Gic: Gaussian-informed continuum for physical property identification and simulation. *Advances in Neural Information Processing Systems*, 37:75035–75063, 2024.

Junyi Cao, Shanyan Guan, Yanhao Ge, Wei Li, Xiaokang Yang, and Chao Ma. Neuma: Neural material adaptor for visual grounding of intrinsic dynamics. *Advances in Neural Information Processing Systems*, 37:65643–65669, 2024.

Hsiao-yu Chen, Edith Tretschk, Tuur Stuyck, Petr Kadlecek, Ladislav Kavan, Etienne Vouga, and Christoph Lassner. Virtual elastic objects. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15806–15816. IEEE, 2022a.

Peter Yichen Chen, Jinxu Xiang, Dong Heon Cho, Yue Chang, GA Pershing, Henrique Teles Maia, Maurizio M Chiaramonte, Kevin Carlberg, and Eitan Grinspun. Crom: Continuous reduced-order modeling of pdes using implicit neural representations. *arXiv preprint arXiv:2206.02607*, 2022b.

Haoyi Duan, Hong-Xing Yu, Sirui Chen, Li Fei-Fei, and Jiajun Wu. Worldscore: A unified evaluation benchmark for world generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025.

Philipp Erler, Paul Guerrero, Stefan Ohrhallinger, Niloy J. Mitra, and Michael Wimmer. Points2surf learning implicit surfaces from point clouds. In *European conference on computer vision*, pp. 108–124. Springer International Publishing, 2020.

Yutao Feng, Yintong Shang, Xuan Li, Tianjia Shao, Chenfanfu Jiang, and Yin Yang. Pie-nerf: Physics-based interactive elastodynamics with nerf. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4450–4461. IEEE, 2024.

Yutao Feng, Xiang Feng, Yintong Shang, Ying Jiang, Chang Yu, Zeshun Zong, Tianjia Shao, Hongzhi Wu, Kun Zhou, Chenfanfu Jiang, et al. Gaussian splashing: Unified particles for versatile motion synthesis and rendering. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 518–529. IEEE, 2025.

Zhoujie Fu, Jiacheng Wei, Wenhao Shen, Chaoyue Song, Xiaofeng Yang, Fayao Liu, Xulei Yang, and Guosheng Lin. Sync4D: Video Guided Controllable Dynamics for Physics-Based 4D Generation. *arXiv preprint arXiv:2405.16849*, 2024.

Moritz Geilinger, David Hahn, Jonas Zehnder, Moritz Bächer, Bernhard Thomaszewski, and Stelian Coros. Add: Analytically differentiable dynamics for multi-body systems with frictional contact. *ACM Transactions on Graphics (TOG)*, 39(6):1–15, 2020.

Chen Geng, Guangzhao He, Yue Gao, Yunzhi Zhang, Shangzhe Wu, and Jiajun Wu. Neurok: Generative 4d neural object kinematics. *arXiv preprint arXiv:2605.30347*, 2026.

Eric Heiden, Miles Macklin, Yashraj Narang, Dieter Fox, Animesh Garg, and Fabio Ramos. Disect: A differentiable simulation engine for autonomous robotic cutting. *arXiv preprint arXiv:2105.12244*, 2021.

Yuanming Hu, Yu Fang, Ziheng Ge, Ziyin Qu, Yixin Zhu, Andre Pradhana, and Chenfanfu Jiang. A moving least squares material point method with displacement discontinuity and two-way rigid body coupling. *ACM Transactions on Graphics (TOG)*, 37(4):1–14, 2018.

Yuanming Hu, Luke Anderson, Tzu-Mao Li, Qi Sun, Nathan Carr, Jonathan Ragan-Kelley, and Frédo Durand. Difftaichi: Differentiable programming for physical simulation. *arXiv preprint arXiv:1910.00935*, 2019.

Tianyu Huang, Haoze Zhang, Yihan Zeng, Zhilu Zhang, Hui Li, Wangmeng Zuo, and Rynson WH Lau. Dreamphysics: Learning physics-based 3d dynamics with video diffusion priors. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 3733–3741, 2025.

Zhiao Huang, Yuanming Hu, Tao Du, Siyuan Zhou, Hao Su, Joshua B. Tenenbaum, and Chuang Gan. Plasticinelab: A soft-body manipulation benchmark with differentiable physics. *arXiv preprint arXiv:2104.03311*, 2021.

Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, Yaohui Wang, Xinyuan Chen, Limin Wang, Dahua Lin, Yu Qiao, and Ziwei Liu. Vbench: Comprehensive benchmark suite for video generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Tomas Jakab, Richard Tucker, Ameesh Makadia, Jiajun Wu, Noah Snavely, and Angjoo Kanazawa. KeypointDeformer: Unsupervised 3d keypoint discovery for shape control. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12783–12792, 2021.

Chenfanfu Jiang, Craig Schroeder, Joseph Teran, Alexey Stomakhin, and Andrew Selle. The material point method for simulating continuum materials. In *Acm siggraph 2016 courses*, pp. 1–52, 2016.

Hanxiao Jiang, Hao-Yu Hsu, Kaifeng Zhang, Hsin-Ni Yu, Shenlong Wang, and Yunzhu Li. Phystwin: Physics-informed reconstruction and simulation of deformable objects from videos. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 7219–7230. IEEE, 2025.

Yanqin Jiang, Li Zhang, Jin Gao, Weiming Hu, and Yao Yao. Consistent4d: Consistent 360 dynamic object generation from monocular video. In *International Conference on Learning Representations*, volume 2024, pp. 51844–51861, 2024.

Takuhiro Kaneko. Improving physics-augmented continuum neural radiance field-based geometry-agnostic system identification with lagrangian particle optimization. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5470–5480. IEEE, 2024.

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4), July 2023. URL https://repo-sam.inria.fr/fungraph/3d-gaussian-splatting/.

Agelos Kratimenos, Jiahui Lei, and Kostas Daniilidis. Dynmf: Neural motion factorization for real-time dynamic view synthesis with 3d gaussian splatting. In *European Conference on Computer Vision*, pp. 252–269. Springer, 2024.

Long Le, Ryan Lucas, Chen Wang, Chuhao Chen, Dinesh Jayaraman, Eric Eaton, and Lingjie Liu. Pixie: Fast and generalizable supervised learning of 3d physics from pixels. *arXiv preprint arXiv:2508.17437*, 2025.

Chenchang Li, Zihao Ai, Tong Wu, Xiaosa Li, Wenbo Ding, and Huazhe Xu. Deformnet: Latent space modeling and dynamics prediction for deformable object manipulation. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 14770–14776. IEEE, 2024a.

Jinxi Li, Ziyang Song, and Bo Yang. Nvfi: Neural velocity fields for 3d physics learning from dynamic videos. *Advances in Neural Information Processing Systems*, 36:34723–34751, 2023a.

Xuan Li, Yi-Ling Qiao, Peter Yichen Chen, Krishna Murthy Jatavallabhula, Ming Lin, Chenfanfu Jiang, and Chuang Gan. Pac-nerf: Physics augmented continuum neural radiance fields for geometry-agnostic system identification. *arXiv preprint arXiv:2303.05512*, 2023b.

Yifei Li, Tao Du, Kui Wu, Jie Xu, and Wojciech Matusik. Diffcloth: Differentiable cloth simulation with dry frictional contact. *ACM Transactions on Graphics (TOG)*, 42(1):1–20, 2022.

Yunzhu Li, Jiajun Wu, Russ Tedrake, Joshua B. Tenenbaum, and Antonio Torralba. Learning particle dynamics for manipulating rigid bodies, deformable objects, and fluids. *arXiv preprint arXiv:1810.01566*, 2018.

Zhengqi Li, Qianqian Wang, Forrester Cole, Richard Tucker, and Noah Snavely. Dynibar: Neural dynamic image-based rendering. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4273–4284. IEEE, 2023c.

Zhengqi Li, Richard Tucker, Noah Snavely, and Aleksander Holynski. Generative image dynamics. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24142–24153. IEEE, 2024b.

Hanwen Liang, Yuyang Yin, Dejia Xu, Hanxue Liang, Zhangyang Wang, Konstantinos N Plataniotis, Yao Zhao, and Yunchao Wei. Diffusion4d: Fast spatial-temporal consistent 4d generation via video diffusion models. *arXiv preprint arXiv:2405.16645*, 2024a.

Hanwen Liang, Dejia Xu, Neel P Bhatt, Hezhen Hu, Hanxue Liang, and Konstantinos N Plataniotis. Comp4d: Compositional 4d scene generation. In *2026 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 3567–3577. IEEE, 2026.

Junbang Liang, Ming Lin, and Vladlen Koltun. Differentiable cloth simulation for inverse problems. In *Advances in neural information processing systems*, volume 32, 2019.

Xiao Liang, Fei Liu, Yutong Zhang, Yuelei Li, Shan Lin, and Michael Yip. Real-to-sim deformable object manipulation: Optimizing physics models with residual mappings for robotic surgery. In *2024 IEEE international conference on robotics and automation (ICRA)*, pp. 15471–15477. IEEE, 2024b.

Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. Magic3d: High-resolution text-to-3d content creation. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 300–309. IEEE, 2023.

Jiajing Lin, Shu Jiang, Qingyuan Zeng, Zhenzhong Wang, and Min Jiang. Visionlaw: Inferring interpretable intrinsic dynamics from visual observations via bilevel optimization. In *International Conference on Learning Representations*, volume 2026, pp. 96128–96165, 2026.

Yuchen Lin, Chenguo Lin, Jianjin Xu, and Yadong Mu. Omniphysgs: 3d constitutive gaussians for general physics-based dynamics generation. In *International Conference on Learning Representations*, volume 2025, pp. 70160–70187, 2025.

Huan Ling, Seung Wook Kim, Antonio Torralba, Sanja Fidler, and Karsten Kreis. Align your gaussians: Text-to-4d with dynamic 3d gaussians and composed diffusion models. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8576–8588. IEEE, 2024.

Chunjiang Liu, Xiaoyuan Wang, Qingran Lin, Albert Xiao, Haoyu Chen, Shizheng Wen, Hao Zhang, Lu Qi, Ming-Hsuan Yang, Laszlo A Jeni, Min Xu, and Yizhou Zhao. Multi-object system identification from videos. In *International Conference on Learning Representations*, volume 2026, pp. 66204–66234, 2026.

Fangfu Liu, Hanyang Wang, Shunyu Yao, Shengjun Zhang, Jie Zhou, and Yueqi Duan. Physics3d: Learning physical properties of 3d gaussians via video diffusion. *arXiv preprint arXiv:2406.04338*, 2024a.

Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9264–9275. IEEE, 2023.

Shaowei Liu, Zhongzheng Ren, Saurabh Gupta, and Shenlong Wang. Physgen: Rigid-body physics-grounded image-to-video generation. In *European Conference on Computer Vision*, pp. 360–378. Springer, 2024b.

Jonathon Luiten, Georgios Kopanas, Bastian Leibe, and Deva Ramanan. Dynamic 3d gaussians: Tracking by persistent dynamic view synthesis. In *2024 International Conference on 3D Vision (3DV)*, pp. 800–809. IEEE, 2024.

Baorui Ma, Zhizhong Han, Yu-Shen Liu, and Matthias Zwicker. Neural-pull: Learning signed distance functions from point clouds by learning to pull space onto surfaces. *arXiv preprint arXiv:2011.13495*, 2020.

Pingchuan Ma, Tao Du, Joshua B. Tenenbaum, Wojciech Matusik, and Chuang Gan. Risp: Rendering-invariant state predictor with differentiable simulation and rendering for cross-domain parameter estimation. *arXiv preprint arXiv:2205.05678*, 2022.

Pingchuan Ma, Peter Yichen Chen, Bolei Deng, Joshua B Tenenbaum, Tao Du, Chuang Gan, and Wojciech Matusik. Learning neural constitutive laws from motion observations for generalizable pde dynamics. In *International Conference on Machine Learning*, pp. 23279–23300. PMLR, 2023.

Himangi Mittal, Peiye Zhuang, Hsin-Ying Lee, and Shubham Tulsiani. Uniphy: Learning a unified constitutive model for inverse physics simulation. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16208–16218. IEEE, 2025.

Keunhong Park, Utkarsh Sinha, Jonathan T. Barron, Sofien Bouaziz, Dan B Goldman, Steven M. Seitz, and Ricardo Martin-Brualla. Nerfies: Deformable neural radiance fields. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2021a. URL `http://arxiv.org/abs/2011.12948v5`.

Keunhong Park, Utkarsh Sinha, Peter Hedman, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Ricardo Martin-Brualla, and Steven M Seitz. Hypernerf: A higher-dimensional representation for topologically varying neural radiance fields. *arXiv preprint arXiv:2106.13228*, 2021b.

Tobias Pfaff, Meire Fortunato, Alvaro Sanchez-Gonzalez, and Peter W Battaglia. Learning mesh-based simulation with graph networks. *arXiv preprint arXiv:2010.03409*, 2020.

Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022.

Albert Pumarola, Enric Corona, Gerard Pons-Moll, and Francesc Moreno-Noguer. D-NeRF: Neural Radiance Fields for Dynamic Scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

Yi-Ling Qiao, Alexander Gao, and Ming Lin. Neuphysics: Editable neural geometry and physics from monocular videos. *Advances in Neural Information Processing Systems*, 35:12841–12854, 2022.

Yiling Qiao, Junbang Liang, Vladlen Koltun, and Ming Lin. Differentiable simulation of soft multi-body systems. In *Advances in Neural Information Processing Systems*, volume 34, pp. 17123–17135, 2021.

Ri-Zhao Qiu, Ge Yang, Weijia Zeng, and Xiaolong Wang. Language-driven physics-based scene synthesis and editing via feature splatting. In *European conference on computer vision*, pp. 368–383. Springer Nature Switzerland, 2024.

Maziar Raissi, Paris Perdikaris, and George E. Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational physics*, 378:686–707, 2019.

Jiawei Ren, Liang Pan, Jiaxiang Tang, Chi Zhang, Ang Cao, Gang Zeng, and Ziwei Liu. Dreamgaussian4d: Generative 4d gaussian splatting. *arXiv preprint arXiv:2312.17142*, 2023.

Daniel Rho, Jun Myeong Choi, Biswadip Dey, and Roni Sengupta. Projo4d: Progressive joint optimization for sparse-view inverse physics estimation. *arXiv preprint arXiv:2506.05317*, 2025.

Remy Sabathier, David Novotny, Niloy J Mitra, and Tom Monnier. Actionmesh: Animated 3d mesh generation with temporal 3d diffusion. *arXiv preprint arXiv:2601.16148*, 2026.

Alvaro Sanchez-Gonzalez, Jonathan Godwin, Tobias Pfaff, Rex Ying, Jure Leskovec, and Peter Battaglia. Learning to simulate complex physics with graph networks. In *International conference on machine learning*, pp. 8459–8468. Pmlr, 2020.

Yidi Shao, Mu Huang, Chen Change Loy, and Bo Dai. Gausim: Registering elastic objects into digital world by gaussian simulator. *arXiv preprint arXiv:2412.17804*, 2024.

Haochen Shi, Huazhe Xu, Samuel Clarke, Yunzhu Li, and Jiajun Wu. Robocook: Long-horizon elasto-plastic object manipulation with diverse tools. *arXiv preprint arXiv:2306.14447*, 2023.

Haochen Shi, Huazhe Xu, Zhiao Huang, Yunzhu Li, and Jiajun Wu. RoboCraft: Learning to see, simulate, and shape elasto-plastic objects in 3D with graph networks. *The International Journal of Robotics Research*, 43(4):533–549, 2024.

Colton Stearns, Adam W. Harley, Mikaela Uy, Florian Dubost, Federico Tombari, Gordon Wetzstein, and Leonidas Guibas. Dynamic gaussian marbles for novel view synthesis of casual monocular videos. In *ArXiv*, 2024.

Alexey Stomakhin, Craig Schroeder, Lawrence Chai, Joseph Teran, and Andrew Selle. A material point method for snow simulation. *ACM Transactions on Graphics (TOG)*, 32(4):1–10, 2013.

Deborah Sulsky, Shi-Jian Zhou, and Howard L. Schreyer. Application of a particle-in-cell method to solid mechanics. *Computer physics communications*, 87(1–2):236–252, 1995.

Tetsuya Takahashi and Ming C. Lin. Video-guided real-to-virtual parameter transfer for viscous fluids. *ACM Transactions on Graphics (TOG)*, 38(6):1–12, 2019.

Jiaxiang Tang, Jiawei Ren, Hang Zhou, Ziwei Liu, and Gang Zeng. Dreamgaussian: Generative gaussian splatting for efficient 3d content creation. In *International Conference on Learning Representations*, volume 2024, pp. 33879–33896, 2024.

Bin Wang, Longhua Wu, KangKang Yin, Uri M. Ascher, Libin Liu, and Hui Huang. Deformation capture and modeling of soft objects. *ACM Trans. Graph.*, 34(4), 2015.

Chaoyang Wang, Lachlan Ewen MacDonald, Laszlo A Jeni, and Simon Lucey. Flow supervision for deformable nerf. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21128–21137. IEEE, 2023a.

Chen Wang, Chuhao Chen, Yiming Huang, Zhiyang Dou, Yuan Liu, Jiatao Gu, and Lingjie Liu. Physctrl: Generative physics for controllable and physics-grounded video generation. *Advances in Neural Information Processing Systems*, 38:167907–167932, 2026.

Jiahong Wang, Yinwei Du, Stelian Coros, and Bernhard Thomaszewski. Neural modes: Self-supervised learning of nonlinear modal subspaces. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 23158–23167. IEEE, 2024.

Zhengyi Wang, Cheng Lu, Yikai Wang, Fan Bao, Chongxuan Li, Hang Su, and Jun Zhu. Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. *Advances in neural information processing systems*, 36:8406–8441, 2023b.

Guanjun Wu, Taoran Yi, Jiemin Fang, Lingxi Xie, Xiaopeng Zhang, Wei Wei, Wenyu Liu, Qi Tian, and Xinggang Wang. 4d gaussian splatting for real-time dynamic scene rendering. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20310–20320. IEEE, 2024.

Rundi Wu, Ruiqi Gao, Ben Poole, Alex Trevithick, Changxi Zheng, Jonathan T Barron, and Aleksander Holynski. Cat4d: Create anything in 4d with multi-view video diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 26057–26068, 2025a.

Zijie Wu, Chaohui Yu, Fan Wang, and Xiang Bai. Animateanymesh: A feed-forward 4d foundation model for text-driven universal mesh animation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025b.

Wenqi Xian, Jia-Bin Huang, Johannes Kopf, and Changil Kim. Space-time neural irradiance fields for free-viewpoint video. 2021 ieee. In *CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9416–9426, 2020.

Tianyi Xie, Zeshun Zong, Yuxing Qiu, Xuan Li, Yutao Feng, Yin Yang, and Chenfanfu Jiang. Physgaussian: Physics-integrated 3d gaussians for generative dynamics. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4389–4398. IEEE, 2024.

Yiming Xie, Chun-Han Yao, Vikram Voleti, Huaizu Jiang, and Varun Jampani. Sv4d: Dynamic 3d content generation with multi-frame and multi-view consistency. In *International Conference on Learning Representations*, volume 2025, pp. 33421–33441, 2025.

Zhenjia Xu, Jiajun Wu, Andy Zeng, Joshua B. Tenenbaum, and Shuran Song. Densephysnet: Learning dense physical object representations via multi-step dynamic interactions. *arXiv preprint arXiv:1906.03853*, 2019.

Zeyu Yang, Hongye Yang, Zijie Pan, and Li Zhang. Real-time photorealistic dynamic scene representation and rendering with 4d gaussian splatting. In *International Conference on Learning Representations*, volume 2024, pp. 9142–9159, 2024a.

Ziyi Yang, Xinyu Gao, Wen Zhou, Shaohui Jiao, Yuqing Zhang, and Xiaogang Jin. Deformable 3d gaussians for high-fidelity monocular dynamic scene reconstruction. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20331–20341. IEEE, 2024b.

Jiraphon Yenphraphai, Jianqi Chen, Jian Wang, Gordon Guocheng Qian, Sergey Tulyakov, Rameen Abdal, Raymond A. Yeh, Peter Wonka, and Chaoyang Wang. Helix4d: Complex 4d mesh generation. *arXiv preprint arXiv:2605.26109*, 2026.

Yuyang Yin, Dejia Xu, Zhangyang Wang, Yao Zhao, and Yunchao Wei. 4dgen: Grounded 4d content generation with spatial-temporal consistency. *arXiv preprint arXiv:2312.17225*, 2023.

Heng Yu, Joel Julin, Zoltan A Milacski, Koichiro Niinuma, and Laszlo A Jeni. Cogs: Controllable gaussian splatting. *arXiv*, 2023a.

Heng Yu, Joel Julin, Zoltan A Milacski, Koichiro Niinuma, and Laszlo A Jeni. Dylin: Making light field networks dynamic. *arXiv preprint arXiv:2303.14243*, 2023b.

Yifei Zeng, Yanqin Jiang, Siyu Zhu, Yuanxun Lu, Youtian Lin, Hao Zhu, Weiming Hu, Xun Cao, and Yao Yao. Stag4d: Spatial-temporal anchored generative 4d gaussians. In *European Conference on Computer Vision*, pp. 163–179. Springer, 2024.

Tianyuan Zhang, Hong-Xing Yu, Rundi Wu, Brandon Y Feng, Changxi Zheng, Noah Snavely, Jiajun Wu, and William T Freeman. Physdreamer: Physics-based interaction with 3d objects via video generation. In *European Conference on Computer Vision*, pp. 388–406. Springer, 2024.

Yizhou Zhao, Haoyu Chen, Chunjiang Liu, Zhenyang Li, Charles Herrmann, Junhwa Hur, Yinxiao Li, Ming-Hsuan Yang, Bhiksha Raj, and Min Xu. Toward material-agnostic system identification from videos. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 5944–5956. IEEE, 2025.

Dongzhe Zheng, Siqiong Yao, Wenqiang Xu, and Cewu Lu. Differentiable Cloth Parameter Identification and State Estimation in Manipulation. *IEEE Robotics and Automation Letters*, 9(3): 2519–2526, 2024.

Licheng Zhong, Hong-Xing Yu, Jiajun Wu, and Yunzhu Li. Reconstruction and simulation of elastic objects with spring-mass 3d gaussians. In *European Conference on Computer Vision*, pp. 407–423. Springer, 2024.

SUPPLEMENTARY MATERIAL

This supplement provides notation and task conventions (Appendix A.1), method derivations (Appendix A), implementation details (Appendix B), and the dataset and evaluation protocol (Appendix C). Additional quantitative and qualitative analyses appear in Appendices D.1 and D.2.

A ADDITIONAL METHOD DETAILS

A.1 TASK SETUP AND NOTATION

Symbols and Coordinate Systems. The reference mesh is $\mathcal{M}_0$, and $\mathbf{x}_0 \in \mathbb{R}^{3n}$ contains its sampled object-local points. Local deformed points are $\mathbf{x}_t$, reduced coordinates are z_t, and $J(z) = \partial\mathcal{F}(z)/\partial z$ is the decoder Jacobian. The global pose consists of $R_t \in \mathrm{SO}(3)$ and translation $\mathbf{r}_t$. The reference normalization uses scale ℓ and center $\mathbf{o}$. Hats denote predictions. Nonrigid point force is $\mathbf{f}_{\mathrm{nr}}$, its latent projection is Q, the fixed observed-response context is c, and the evolving forecast-network state is h_k. Positions are converted back to SI units before reporting millimeter error. Subscript t indexes stored observation frames; k indexes integration microsteps. Dots denote derivatives with respect to physical time τ, with $\tau_{k+1} - \tau_k = \Delta t$, rather than differences in the integer index.

Formal Prediction Task. For each task, the predictor receives a reference shape, one complete observed demonstration $\mathcal{D}$, a separately specified query condition $\mathcal{B}^q$, and the query state at frame zero. It encodes $\mathcal{D}$ once and predicts query deformation and global pose for frames $1{:}T_q$. Cross-condition tasks pair distinct interactions with the same geometry and constitutive parameters. Same-trajectory pairs provide auxiliary reconstruction supervision. Training and validation are disjoint by geometry identity.

A.2 DEFORMATION REPRESENTATION

Rigid–Nonrigid Factorization. For corresponding world-space points, we estimate a per-frame rigid transform relative to the canonical reference and express residual deformation in the reference local frame. Applying Equation 2 recomposes the two components. We retain sequences with stable rigid alignment and correspondence-consistent local residuals.

Frozen Decoder and Rigid Gauge. Let $D_{\mathrm{raw}}(u; \mathcal{M}_0)$ be the finetuned geometry-conditioned decoder and μ_{cond} its reference code. For compact coordinates z and basis B, first decode

$$\widetilde{\mathbf{x}}(z) = D_{\mathrm{raw}}(\mu_{\mathrm{cond}} + Bz; \mathcal{M}_0). \tag{9}$$

The effective local decoder is

$$\mathcal{F}(z; \mathcal{M}_0) = \Gamma_{\mathcal{M}_0}(\widetilde{\mathbf{x}}(z)), \tag{10}$$

where $\Gamma_{\mathcal{M}_0}$ removes the centroid and applies the best-fitting proper rotation into the reference frame. This differentiable gauge fixes global translation and rotation while preserving the decoded nonrigid deformation.

Geometry-Specific Compact Basis. We sample latent codes near μ_{cond} and differentiate decoded points with respect to the full 131,072-dimensional code. For each sampled Jacobian, infinitesimal translation is removed by centering point displacement, and infinitesimal rotation is removed by least-squares projection onto the three rotational velocity fields about the reference center. Stacking the residual Jacobians and retaining their leading right singular vectors yields an orthonormal active subspace $B \in \mathbb{R}^{131072 \times r}$. We use $r = 20$ in the main model.

Decoder Jacobian in Compact Coordinates. The full code and reduced coordinates are related by $u = \mu_{\mathrm{cond}} + Bz$. With the representation and B fixed, the chain rule gives

$$J(z) = \left.\frac{\partial \Gamma_{\mathcal{M}_0}}{\partial \widetilde{\mathbf{x}}}\right|_{\widetilde{\mathbf{x}}(z)} \left.\frac{\partial D_{\mathrm{raw}}(u; \mathcal{M}_0)}{\partial u}\right|_{u=\mu_{\mathrm{cond}}+Bz} B. \tag{11}$$

Thus the reduced Jacobian has 20 columns even though the full code has 131,072 components. The derivative includes the rigid-gauge operation used in the effective decoder. In particular, orthonormality of the latent basis, $B^\mathsf{T}B = I$, does not imply $J(z)^\mathsf{T}J(z) = I$ in point space. For a time-varying reduced state,

$$\dot{\mathbf{x}} = J(z)\dot{z}, \qquad \ddot{\mathbf{x}} = J(z)\ddot{z} + \dot{J}(z)\dot{z}. \tag{12}$$

The predicted latent acceleration therefore determines a decoded acceleration through both the current Jacobian and the curvature of the deformation map.

A.3 Coordinate and Velocity Fitting

Coordinate Fitting. For each trajectory, full latent codes are initialized from the posterior mean and optimized for 200 Adam steps at learning rate 0.1 while the representation remains frozen. Rank-20 coordinates are initialized by least-squares projection of the fitted full latent residual and optimized for 200 steps against the frozen full-latent decoded surface:

$$\mathcal{L}_{\text{fit}} = \frac{1}{Tn}\sum_{t,i} \left\| \mathcal{F}(z_t)_i - \mathbf{x}_{t,i}^{\text{full}} \right\|_2^2 . \tag{13}$$

The causal surface secant between frames is then projected through the current Jacobian to recover $\dot{z}_t$ by regularized least squares. These fitted states provide dynamics supervision.

Causal Velocity Fitting. For $t \geq 1$, let $\delta\tau_t$ be the physical duration between consecutive stored frames and let $\mathbf{x}_t$ denote the fitted local surface in the shared reference frame. Its backward secant is $\mathbf{v}_t^{\text{sec}} = (\mathbf{x}_t - \mathbf{x}_{t-1})/\delta\tau_t$. The regularized least-squares problem underlying the velocity projection is

$$\dot{z}_t = \arg\min_{v \in \mathbb{R}^{20}} \|J(z_t)v - \mathbf{v}_t^{\text{sec}}\|_2^2 + \lambda\|v\|_2^2, \tag{14}$$

$$(J_t^\mathsf{T}J_t + \lambda I)\dot{z}_t = J_t^\mathsf{T}\mathbf{v}_t^{\text{sec}}, \qquad J_t = J(z_t). \tag{15}$$

Here $\lambda > 0$ denotes the regularization strength. Only frames t and $t-1$ enter this velocity estimate. The frame-zero reduced velocity follows the declared initial condition; in the reported query benchmark it is initialized to zero. This observation interval is distinct from the smaller integration microstep Δt.

A.4 Generalized Forces and Integration

Force Conversion and State Updates. Point force is transformed to the decoder-local frame and stripped of instantaneous rigid force and torque components. Its pullback $Q = J(z)^\mathsf{T}\mathbf{f}_{\text{nr}}$ is recomputed at every microstep. Coordinates are advanced with Equation 7; global pose is advanced causally from the predicted surface state and known forces and torques, and is recomposed only after local decoding.

Virtual-Work Interpretation. At fixed global pose, a virtual displacement of the local decoded surface is $\delta\mathbf{x} = J(z)\delta z$. For the nonrigid point-force component expressed in the corresponding work-conjugate local coordinates,

$$\delta W = \mathbf{f}_{\text{nr}}^\mathsf{T}\delta\mathbf{x} = \mathbf{f}_{\text{nr}}^\mathsf{T}J(z)\delta z = Q^\mathsf{T}\delta z, \qquad Q = J(z)^\mathsf{T}\mathbf{f}_{\text{nr}}. \tag{16}$$

This is the generalized-force pullback used with learned deformation coordinates in NeuROK (Geng et al., 2026). Normalized local coordinates require a consistent force conversion: from Equation 2, a point force in SI world coordinates has work-conjugate local force $\ell R^\mathsf{T}\mathbf{f}^{\text{world,SI}}$ when the rigid pose is held fixed. Rigid force and torque are handled by the global-motion update.

Force Features and Acceleration. The pullback is a generalized force, not a latent acceleration. If M is the point-mass matrix and J_{SI} is the local decoder Jacobian expressed in physical lengths, the associated kinetic energy and reduced mass matrix are

$$T_{\text{local}} = \tfrac{1}{2}\dot{z}^\mathsf{T}M_z(z)\dot{z}, \qquad M_z(z) = J_{\text{SI}}(z)^\mathsf{T}MJ_{\text{SI}}(z). \tag{17}$$

Our conditional predictor learns the acceleration response to Q, geometry, and observed-response context rather than identifying Q with $\ddot{z}$. The mass expression explains the force semantics; it does not introduce an additional analytic mass-matrix solve into the reported model.

Numerical Time Convention. Semi-implicit Euler first updates velocity and then uses that updated velocity for position, as in Equation 7. Four integration microsteps cover one observation interval. The index increments by $k \leftarrow k + 1$, whereas physical time increments by $\tau \leftarrow \tau + \Delta t$. The force, contact, and decoder state are updated at each microstep; the demonstration context stays fixed for the complete query.

B Additional Implementation Details

B.1 Representation Learning

Architecture and Objective. The conditional VAE uses four motion-encoder blocks, four decoder blocks, 16 attention heads, 512 sampled surface points, and a 128×1024 full latent. It predicts one 8D dual quaternion per sampled point. Its production objective is

$$\mathcal{L}_{\mathrm{VAE}} = \mathrm{MSE}(\widehat{\mathbf{T}}, \mathbf{T}) + 0.01 D_{\mathrm{KL}}(q_{\mathcal{M}_0} \| p_{\mathcal{M}_0}), \tag{18}$$

where $\mathbf{T}$ denotes the per-point dual-quaternion target and $p_{\mathcal{M}_0} = \mathcal{N}(\mu_{\mathrm{cond}}, I)$.

Finetuning Schedule. Representation finetuning uses AdamW with learning rate 8×10^{-5}, weight decay 8×10^{-6}, batch size 16 per GPU, cosine warm restarts, and adaptive gradient clipping capped at 1.

B.2 Demonstration-Conditioned Dynamics

Network Architectures. The demonstration network is either a GRU or a four-block, four-head causal Transformer with rotary temporal position encoding; both produce a 32-dimensional context, and the Transformer reads its final token. The query network is either a 128-dimensional GRU or a four-block, four-head causal Transformer with RoPE whose 256-microstep window spans 64 observation intervals. A 1024-dimensional geometry code is projected to 32 dimensions. The output module contains a learned bias-free external branch gated by geometry and a learned internal branch with zero-initialized adaptive LayerNorm; both readouts start at zero. Global pose is updated by causal rigid-body integration. We train exactly three demo–query variants, all with learned response conditioning and AdaLN-Zero: GRU–GRU, Causal Transformer–GRU, and Causal Transformer–Causal Transformer (Table 3). The model-selection protocol uses the complete validation rollout.

Pair Construction and Optimization. A cross-condition pair matches split, geometry, rest shape, and constitutive parameters while using different interaction types; self-pairs provide reconstruction auxiliaries. Dynamics coordinates and velocities are standardized per dimension; force, pose, and net force/torque features use frozen training-set scales and an asinh transform. The dynamics optimizer is Adam with learning rate 10^{-3}, no learning-rate scheduler, global batch size 128 per architecture, and gradient clipping at 1. Each architecture is optimized independently. The rollout horizon is 4, 8, and 16 frames for successive 5,000-sample stages and becomes the complete query trajectory after 15,000 samples seen. Dynamics training backpropagates through recurrent rollout, semi-implicit integration, predicted-state contact, and the frozen decoder using Equation 8. The model-selection protocol uses complete geometry-disjoint query validation, with cross-condition and reconstruction metrics reported separately.

Inference Procedure. At test time we (1) fit the complete demonstration in the frozen coordinate system, (2) encode its coordinates, velocities, and generalized forces into c, (3) initialize a separate query from its frame-zero state, (4) recurrently predict local acceleration and causally integrate global pose, and (5) decode and compose the complete world-space query. Network parameters remain fixed during inference.

C Additional Experimental Details

C.1 Data Preparation

Representation and MPM Corpora. Representation finetuning uses 11,137 training clips from 6,695 geometries and 619 validation clips from 379 disjoint geometries, combining DynamicOb-

javerse, DyMesh, and generated MPM sequences. The audited dynamics corpus contains 9,100 generated MPM trajectories from 4,520 geometries. Its geometry-disjoint split contains 8,627 training trajectories from 4,281 geometries and 473 validation trajectories from 239 geometries. The four material families are elastic, plastic, sand, and snow; the five interaction types are drop, press, jet pair, shear, and uniform gust.

Each trajectory stores its material parameters, boundary condition, external-field schedule, particle resolution, frame clock, reference geometry, and random seed. Preprocessing verifies the unrestricted fitted latent, active-subspace provenance, rank-20 coordinates, causal velocities, frame count, and finite values before those states are used for dynamics supervision. Generated queries start from the simulated frame-zero rest state with zero prescribed linear and angular velocity. The rollout uses the fitted frame-zero deformation coordinates and rigid pose and initializes reduced velocity to zero.

Correspondence and Preprocessing. We sample 512 points on the canonical surface, store their face indices and barycentric coordinates, and reuse those identities throughout each topology-preserving trajectory. Representation finetuning follows NeuROK's unified-animation convention: one bounding box over the complete clip, enlarged by 1.05, is shared by every sampled frame pair. Geometry-specific basis construction and dynamics fitting instead use the canonical rest-shape bounding box, again with a 1.05 margin, and apply that single transform to every frame of the trajectory. Rigid alignment, local deformation, force conversion, frame times, and invalid-sequence filters are computed once and stored with the trajectory metadata.

C.2 Evaluation Protocol

Evaluation Splits. Whole geometry identifiers are assigned to train or held-out validation before constructing frame or trajectory pairs. The zero-shot representation benchmark uses target poses from held-out validation geometries. The same-object adaptation benchmark additionally provides its declared support poses for fitting a per-object basis. Dynamics training and normalization use training geometries, while validation uses fixed demonstration–query pairs from disjoint geometries.

For representation evaluation, each validation object has a specified input pose and a distinct target pose. All methods use the same 512 corresponding surface points. We report each method's support set, optimization budget, initialization, and target-fitting objective.

The reported dynamics benchmark contains 79 geometry-disjoint query cases. Sixty-two use an existing trajectory with identical geometry and constitutive parameters but a different loading condition as the demonstration; 17 use a newly simulated demonstration under the same pairing contract. Every trajectory has 32 frames sampled at 24 Hz. The set contains 12 elastic, 24 plastic, 26 sand, and 17 snow queries. Because this set is not uniformly balanced, the aggregate gives every query equal weight rather than averaging material-family means. The post-hoc subset selection is described in Appendix C.3.

For dynamics evaluation, the conditional dynamics models receive the reference geometry, a complete demonstration with its loading, the query frame-zero state, and the query-condition parameters. Baselines that cannot consume all of these inputs receive the method-specific subset disclosed below. Every method produces a complete world-space query trajectory, and causal contact models evaluate contact on the evolving predicted geometry and velocity.

Error Definitions and Reporting. For T predicted frames and n corresponding points, the primary world-space error is

$$E_{\text{world}} = 10^3 \sqrt{\frac{1}{Tn} \sum_{t=1}^{T} \sum_{i=1}^{n} \|\hat{\mathbf{x}}_{t,i}^{\text{world,SI}} - \mathbf{x}_{t,i}^{\text{world,SI}}\|_2^2} \quad \text{mm}. \tag{19}$$

End-point RMSE uses the same definition at $t = T$. Chamfer-L1 and Chamfer-L2 use a fixed surface sample and the stated distance normalization. We aggregate squared distances over points and frames before taking the square root. Benchmark sizes, frame counts, optimization budgets, and available conditioning are specified in this appendix so that the aggregate numbers in the main tables have an explicit evaluation contract.

C.3 Baseline and Ablation Settings

Representation. Released NeuROK and our finetuned checkpoint share point correspondence, preprocessing, and the target-fitting objective. The released checkpoint is step 23,000 and ours is the validation-selected finetuned step 150,000. Both start from their deterministic posterior mean and optimize the unrestricted 128×1024 latent with Adam for 1,000 updates at learning rate 0.1 using corresponding-point MSE.

NeuROK evaluates NeuralDeformationGraphs (NDG) (Božič et al., 2021) with five same-object support poses. We reproduce its SDF, graph, and local multi-MLP pipeline with 500 graph updates and 500 local-shape updates, followed by 1,000 target-fitting updates to graph-node positions and rotations.

The KeypointDeformer adapter uses 12 deterministic farthest-point surface keypoints and a 42-vertex icosphere cage. Fixed local inverse-distance influences map keypoint translations to cage vertices, and mean-value coordinates transfer cage motion to the surface. Target keypoint positions are optimized for 500 Adam updates at learning rate 0.1. The public repository commit is `082d3f1`.

The benchmark contains 50 geometry-disjoint validation clips selected by deterministic SHA-256 ordering: 39 generated MPM, nine DynamicObjaverse, one DyMesh-16, and one DyMesh-32 clip. Each contributes its maximum-temporal-separation pair. We restore each object's physical scale before reporting millimeters and remove the best-fitting proper rigid transform for shape metrics.

Dynamics Evaluation. Tables 2 and 3 report the same 79-query subset. This post-hoc sensitivity analysis omits zero-based query index 8 (jet-pair), which showed large positional drift; no implementation bug has yet been established for that case. The exclusion is applied to every baseline. Each trajectory contributes its mean squared corresponding-point error with equal weight; we take the square root after averaging across trajectories. Our three variants use the same geometry-response indices, with up to 512 original vertices per object. The main table reports the causal-demo/GRU-query variant, which has the lowest world and end-point errors on this subset. Baseline-specific surface sampling is described below.

Dynamics. The reported dynamics evaluation uses 79 query trajectories and scores frames 1–31 after frame-zero initialization. Its available material-family counts are 12 elastic, 24 plastic, 26 sand, and 17 snow trajectories. We aggregate squared corresponding-point errors over each trajectory and then report the case-equal root mean square across the 79 queries. Static prediction retains the query frame-zero state. Query-velocity extrapolation advances its surface and pose with the frame-zero velocities; all frozen queries declare zero initial velocity, so it coincides with static prediction on this benchmark. Demonstration replay transfers the observed motion to the query clock. The projected-force NeuROK adaptation initializes the released step-23,000 representation from the query frame-zero posterior mean, optimizes the full latent for 200 Adam steps at learning rate 0.1, and then fits a rank-20 state for 200 steps at the same learning rate. At rollout time it projects predicted-state external and contact forces through the current decoder Jacobian, removes rigid motion from the local update, and advances global pose with the causal SE(3) integrator using four microsteps per observed frame.

The GNS entry is a surface-graph adaptation checkpoint after 10,000 optimizer updates on the training split. It uses a complete observed demonstration to form graph-level motion and load statistics, starts the independent query from frame zero, and performs a causal 32-frame rollout without reading future query states. Evaluation uses a deterministic subset of corresponding surface vertices, and both GNS and its static reference are scored on exactly those indices.

The AnimateAnyMesh adaptation uses the official 16-frame checkpoint at repository commit `a0e1d82`. It receives the static query frame-zero mesh and one fixed text prompt determined by the drop, jet-pair, press, shear, or uniform-gust interaction class. We retain original vertex correspondence, use deterministic Euclidean farthest-point sampling only above 2,048 vertices, construct local k-nearest-neighbor adjacency for a sampled mesh, run with seed 666, linearly resample the 16 generated frames to the 32-frame query clock, and restore the input scale and initial world pose. It receives no demonstration, numeric force, contact state, or future query frame. This makes it a text-conditioned motion plausibility baseline rather than a material-identification method.

Table 4: Available conditioning, optimization targets, and evaluation role for each baseline family.

Family	Available Information	Evaluation Role
Released / finetuned NeuROK	Reference pose; target pose as fitting objective	Zero-shot representation fitting
NDG reproduction	Five extra same-object poses; target pose as fitting objective	Same-object representation adaptation
KeypointDeformer adapter	Reference pose; target keypoints as fitting objective	Cross-category cage/keypoint fitting
Static / velocity extrapolation	Query frame-zero state	Analytic dynamics baselines
Demonstration replay	Demo and query frame-zero state	Motion-transfer baseline
Three temporal variants (Table 3)	Demo, query frame zero, query condition	Learned conditional dynamics
GNS adaptation	Demo-derived graph state, query frame zero, query condition	Graph dynamics comparison
NeuROK projected-force	Query frame zero and query condition	Causal rank-20 decoder-Jacobian force projection
AnimateAnyMesh adaptation	Query frame-zero mesh and fixed interaction-class text prompt	Text-conditioned motion plausibility

Temporal-Architecture Ablation Protocol. The three temporal-architecture variants in Table 3 share the demonstration encoder inputs, geometry embedding, generalized-force projection, state variables, decoder, contact model, semi-implicit integrator, training data, and loss. All three use learned response contexts and AdaLN-Zero; only the demonstration encoder and query predictor vary.

D Additional Results

D.1 Quantitative Analysis

Representation Error Variability. Table 5 gives the mean and standard deviation corresponding to the held-out target-pose fitting comparison in Table 1. The support inputs and fitting budgets are specified in Appendix C.3.

Table 5: Target-pose fitting variability on held-out geometries (mean $\pm$ standard deviation; lower is better). The evaluation protocol and methods match Table 1.

Method	Extra Support Poses	NR-RMSE	Chamfer-L1	Chamfer-L2	P90
NDG reproduction (Božič et al., 2021)	5	–	.0328±.0342	.0472±.0470	.0758±.0867
KeypointDeformer adapter (Jakab et al., 2021)	0	.0297±.0221	.0243±.0161	.0301±.0201	.0820±.0622
NeuROK (Geng et al., 2026)	0	.0122±.0075	.0135±.0065	.0155±.0075	.0313±.0200
G-Inverse (Ours)	0	**.0115**±.0082	**.0124**±.0062	**.0142**±.0073	**.0296**±.0220

Full-Latent and Compact Fitting Diagnostics. Figure 3 examines the deformation representation on one held-out fish trajectory. It separates local deformation from global pose, illustrates the geometry-specific basis, and compares unrestricted and rank-20 fits to the same MPM target. Across the 32 frames, the reported RMSE values are 3.02 mm and 4.96 mm, respectively. The accompanying curves show framewise reconstruction error and the optimization trace of the full-latent fit. These are target-fitting diagnostics: the trajectory is available during fitting. They measure representation and compression error separately from unseen-query rollout error and do not constitute a rank sweep or a dynamics-generalization experiment.

D.2 Qualitative Analysis

Trajectory Reconstruction. Figure 4 shows strawberry and bread reconstruction examples at three timestamps. The strawberry illustrates residual plastic deformation after a drop; the bread illustrates elastic deformation under symmetric plate compression. The same object texture and matched timestamps facilitate visual comparison with the simulated trajectory. These reconstruction

examples complement the changed-condition apple comparison in Figure 5; they are not evidence of transfer to an unseen loading condition by themselves.