COSSERATGEN: PHYSICS-GROUNDED VIDEO GENERATION FOR SLENDER DEFORMABLE OBJECTS

Anonymous authors
Paper under double-blind review

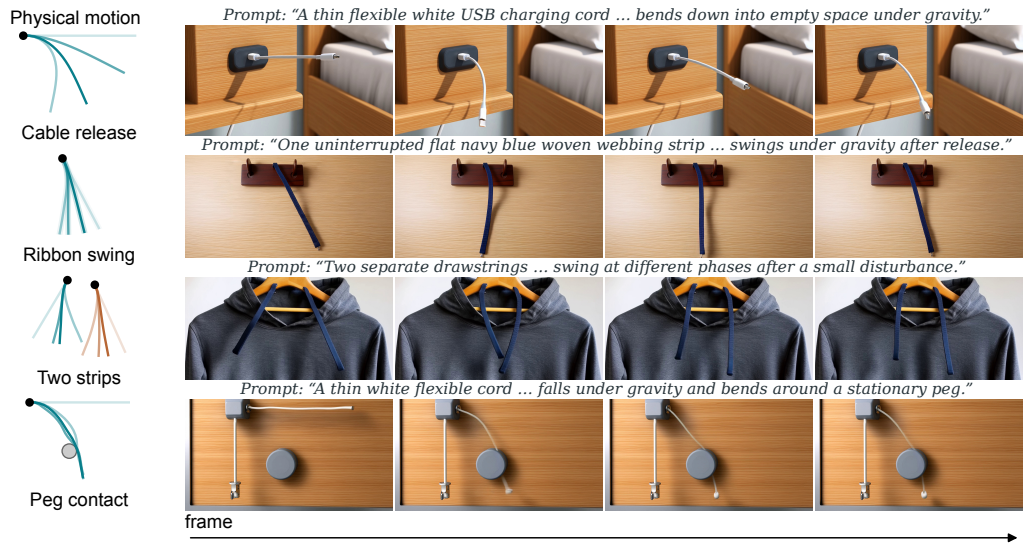

Figure 1: CosseratGen generates videos of slender objects undergoing release, swing, and contact in everyday scenes. Each row pairs a projected trajectory (left) with four generated frames (right) under the displayed prompt. Darker traces indicate later states; black dots mark fixed endpoints.

ABSTRACT

Controlling deformation in generated videos remains challenging for slender objects such as cables, hoses, and ribbons. Specifying an object's path does not determine how its shape should respond to material properties, loads, and constraints. The underlying challenge is to model the distributed dynamics that govern bending and rotation along the object. We introduce CosseratGen, a framework that combines learned slender body dynamics with pretrained video generation. At its core, the Direct-Time Neural Cosserat Operator (DT-NCO) learns from Cosserat rod simulations to predict centerlines and material frames given initial states and prescribed physical conditions, at requested times within the learned horizon. Material frames capture the orientation of local cross sections, distinguishing anisotropic bending responses that the initial centerline alone cannot determine. Projected predictions guide a frozen video generator, allowing physical conditions to control deformation while the scene prompt specifies appearance. Experiments demonstrate improved centerline prediction on withheld mechanics compositions and improved structural adherence in generated videos. By combining learned dynamics with generative priors, CosseratGen enables control over slender object deformation through explicit physical parameters.

1 INTRODUCTION

Recent advances in video generation enable visually realistic scenes and increasing control over object motion (Ho et al., 2022; Wan et al., 2025; Zhang et al., 2025b). However, controlling where

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.

an object moves does not determine how it deforms under physical conditions. This distinction is important for slender objects such as cables, hoses, and ribbons, whose motion involves bending along their length. For example, a charging cable pulled around an organizer should bend according to its material and the applied force while remaining attached to its plug. A visually smooth video can instead show the cable shortening, detaching, or passing through the support. Our goal is to connect the deformation of slender objects in generated videos to the physical conditions that drive their motion.

The difficulty lies in predicting deformation, rather than prescribing its visible path. Trajectory and flow controls specify how image regions should move (Li et al., 2025b; Namekata et al., 2025; Burgert et al., 2025), but a prescribed path does not determine how that motion should change when the material or load changes. Recent methods address this distinction by learning physical dynamics to guide video synthesis (Gillman et al., 2026; Wang et al., 2026; Yuan et al., 2026). NewtonGen, for example, predicts the position, orientation, and shape attributes of an object. For slender objects, the response also depends on how individual segments bend and rotate along the body. Modeling this response therefore requires a distributed physical state that relates local deformation to material properties, applied loads, and boundary conditions.

The initial centerline does not fully specify the deformation response of an anisotropic slender object. Two ribbons can have the same initial centerline yet bend differently under the same load when their cross sections have different orientations. Material frames describe this local orientation, complementing the shape represented by the centerline. Cosserat rod mechanics describes their coupled evolution, and its simulated solutions provide supervision for learning responses to different physical conditions. This motivates our central question: can a shared model learn distributed dynamics from these states and use its predictions to control deformation in generated videos?

In this paper, we introduce CosseratGen, a framework that combines learned slender body dynamics with a pretrained video generator. Its core is the Direct-Time Neural Cosserat Operator (DT-NCO), which learns from simulator states to predict centerlines and material frames given an initial state and prescribed physical conditions. The operator directly predicts configurations at requested times within the learned horizon, with geometric constraints preserving valid rotations and fixed attachments. We project the predicted states into structural guides and use a frozen video generator to synthesize videos conditioned on these guides and the scene prompt (Jiang et al., 2025; Wan et al., 2025). Learning the dynamics also allows the predictor to be adapted to measured motion and physical controls to be optimized through its gradients. This separates learning and controlling deformation from synthesizing its appearance, while allowing each stage to be evaluated independently.

Our contributions are threefold:

- We introduce CosseratGen, a framework that connects prescribed physical conditions to the deformation of slender objects in generated videos through learned dynamics and structural guidance.
- We develop DT-NCO to learn distributed dynamics from simulator supervision, using centerlines and material frames to predict deformation across physical conditions at requested times within a learned horizon.
- We construct Cosserat-12K, a trajectory dataset with distributed state labels and physical conditions, to support learning and evaluation on unseen instances and held-out combinations of the modeled mechanics. Experiments show improved deformation prediction and structural adherence in generated videos, with additional evidence for adaptation to real motion and differentiable physical control.

2 Related Work

Video Generation and Motion Control. Pretrained video models synthesize realistic appearance and motion (Ho et al., 2022; Yang et al., 2025; Wan et al., 2025; Wu et al., 2025; Xing et al., 2024; Girdhar et al., 2024). Their motion can be steered by compositional signals, sparse features, trajectories, masks, points, or boxes (Wang et al., 2023; Guo et al., 2024; Burgert et al., 2025; Zhang et al., 2025b; Li et al., 2025b; Namekata et al., 2025; Yariv et al., 2025; Shi et al., 2025). VACE accepts reference images, masks, and spatiotemporal inputs (Jiang et al., 2025); we therefore use

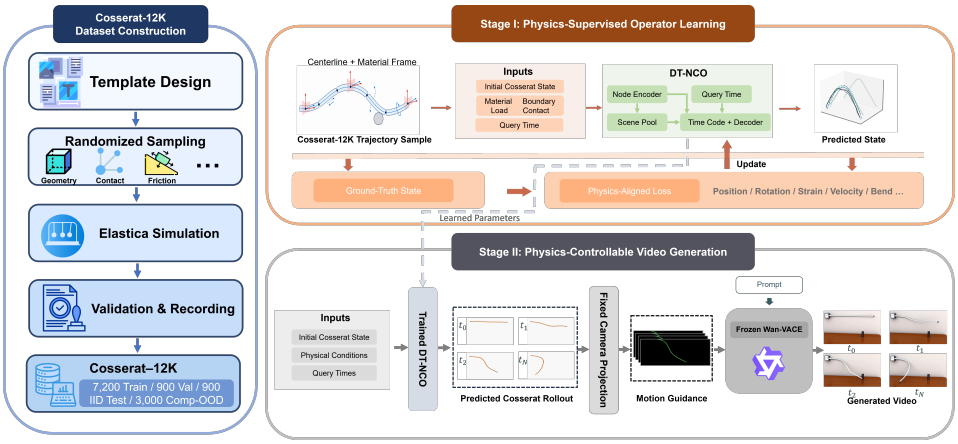

Figure 2: Overview of CosseratGen. Left: Elastica simulates randomized rod configurations to construct Cosserat-12K. Top: simulator states supervise DT-NCO, which predicts centerlines and material frames from the initial state, physical conditions, and query time. Bottom: predicted configurations are projected into structural guides for frozen Wan-VACE, which synthesizes a video under the scene prompt.

frozen Wan-VACE for renderer-matched ablations. These methods specify *where* visible regions move, whereas CosseratGen predicts distributed deformation under prescribed material, boundary, loading, and contact conditions before rendering.

Physics-Grounded Generative Models. Physics-aware generation uses force, particle, and dynamics guidance (Gillman et al., 2026; Wang et al., 2026; Yuan et al., 2026; Li et al., 2025a; Foo et al., 2026; Xie et al., 2024), explicit or learned simulation (Liu et al., 2024; Zhang et al., 2024; 2025a), latent dynamics (Satish et al., 2026; Shen et al., 2026), or pixel-aligned controls (Narayanan et al., 2026). We separate mechanics from appearance through distributed centerlines and material frames, distinguishing identical centerlines with different bending responses.

Learned Simulation of Deformable Objects. Graph simulators learn autoregressive updates (Sanchez-Gonzalez et al., 2020; Pfaff et al., 2020), while Neural ODEs integrate learned dynamics (Chen et al., 2018). Rod networks predict constraint-solver initializations (Shao et al., 2021) or infer loads from shapes (Tang et al., 2025); deformable linear object representations also support planning (Envall et al., 2025). Our operator predicts states directly at query times, using rod-mechanics states (Antman, 2005; Gazzola et al., 2018; Bergou et al., 2008) and Elastica supervision (Naughton et al., 2021). State accuracy and boundary-constraint errors complement video benchmarks (Huang et al., 2024; Bansal et al., 2025; Meng et al., 2024; Li et al., 2026).

3 Method

3.1 Problem Formulation and Overview

CosseratGen separates predicting physical deformation from synthesizing its appearance (Figure 2). The dynamics input is a numerical mechanics specification $\mathcal{P} = (\mathcal{S}_0, \mathbf{a})$, where $\mathcal{S}_0 = (\mathbf{r}_0, \mathbf{v}_0, \mathbf{Q}_0, \boldsymbol{\omega}_0, \boldsymbol{\sigma}_0, \boldsymbol{\kappa}_0)$ contains initial positions, velocities, material frames, angular velocities, and strains. The attributes $\mathbf{a}$ specify material, boundary, loading, driving, and environment conditions. These quantities are supplied explicitly; the framework does not infer them from a sentence or an image. A separate prompt p describes the visual scene. It affects video synthesis but leaves the predicted 3D rollout unchanged.

For physical query times $0 = t_0 < \cdots < t_N = T$, the two stages are

$$\begin{aligned} \widehat{\mathcal{X}}_j &= \mathcal{F}_\theta(\mathcal{P}, t_j) = (\widehat{\mathbf{r}}_{t_j}, \widehat{\mathbf{Q}}_{t_j}), \quad j = 0, \ldots, N, \\ (\mathbf{H}, \mathbf{M}) &= \Pi(\widehat{\mathcal{X}}_{0:N}; C), \qquad \widehat{\mathcal{V}}_{0:N} = G_\psi(\mathbf{H}, p; \eta). \end{aligned} \tag{1}$$

Here $\mathcal{F}_\theta$ is DT-NCO, C fixes the projection and framing, and Π rasterizes structural guides $\mathbf{H}$ and corresponding masks $\mathbf{M}$. The renderer settings η include the diffusion seed and protocol-specific conditions, detailed in Appendix K. We train only $\mathcal{F}_\theta$; Wan-VACE G_ψ remains frozen (Jiang et al., 2025; Wan et al., 2025). The output is a video; the intermediate rollout contains positions and frames, with velocities and strains derived for supervision. Playback rate does not change the physical query interval from 0 to T.

3.2 Cosserat State and Simulator Supervision

The centerline alone does not specify local cross-sectional orientation. Following Cosserat rod mechanics (Antman, 2005; Gazzola et al., 2018), let s denote reference arc length, with $0 \le s \le L$, and represent a rod by $\mathbf{r}(s,t) \in \mathbb{R}^3$ and $\mathbf{Q} = [\mathbf{d}_1, \mathbf{d}_2, \mathbf{d}_3]^\top \in \mathrm{SO}(3)$. The director rows are expressed in world coordinates, so $\mathbf{Q}$ maps world vectors into material coordinates. With $\mathbf{e}_3 = (0,0,1)^\top$, the material shear/stretch and bend/twist strains are

$$\boldsymbol{\sigma} = \mathbf{Q}\,\partial_s \mathbf{r} - \mathbf{e}_3, \qquad \boldsymbol{\kappa} = \mathrm{ax}\big(\mathbf{Q}\,\partial_s \mathbf{Q}^\top\big)\,, \tag{2}$$

where ax inverts the skew-matrix map $[\cdot]_\times$. Linear velocity is $\mathbf{v} = \partial_t \mathbf{r}$ and material angular velocity is $\boldsymbol{\omega} = \mathrm{ax}(\mathbf{Q}\partial_t \mathbf{Q}^\top)$. The spatial momentum balances are

$$\begin{aligned} \partial_t(\rho A \mathbf{v}) &= \partial_s \mathbf{n} + \mathbf{f}, \\ \partial_t \boldsymbol{\ell} &= \partial_s \mathbf{m} + \partial_s \mathbf{r} \times \mathbf{n} + \boldsymbol{\tau}, \end{aligned} \tag{3}$$

Here, ρA is mass per reference length; $\mathbf{n}, \mathbf{m}$ are internal force and moment; $\mathbf{f}, \boldsymbol{\tau}$ are distributed external force and torque; and $\boldsymbol{\ell}$ is angular-momentum density, with all vectors expressed in world coordinates. Elastica integrates these balances with constitutive laws, damping, and contact to supervise DT-NCO (Naughton et al., 2021); the surrogate does not explicitly enforce the balance equations.

Cosserat-12K uses isotropic circular rods, whose bending stiffness is invariant to cross-sectional rotation. We therefore test orientation-dependent bending on anisotropic ribbons with identical initial centerlines but different material frames (Appendix D).

Full-frame input to the dynamics predictor supplies all three initial directors; *full-state input* additionally supplies angular velocity and strains. Both retain position and linear velocity. Initial strains are derived features, not independent controls; the output remains a configuration.

3.3 Direct-Time Neural Cosserat Operator

Physical Attributes and Node Encoding. The attributes $\mathbf{a}$ encode material properties, fixed nodes, gravity, contact geometry, and prescribed controls. Factorized encoders combine structural, environmental, and loading attributes without a template identity. A shared obstacle encoder aggregates node-relative cylinder geometry, while the plane is encoded separately. Appendix B specifies all channels and aggregation weights.

For each node (k,i), a shared MLP encodes its initial state, local geometry, physical attributes, material coordinates, and boundary indicator into $\mathbf{z}_{ki}$. Masked scene pooling and a time-conditioned decoder give

$$\bar{\mathbf{z}} = \frac{\sum_{k,i} m_{ki}\mathbf{z}_{ki}}{\sum_{k,i} m_{ki}}, \qquad (\mathbf{u}_{ki}, \mathbf{q}_{ki}) = D_\theta[\mathbf{z}_{ki}, \bar{\mathbf{z}}, \gamma(t)], \tag{4}$$

where m_{ki} excludes padding and $\gamma(t)$ encodes elapsed seconds. Pooling exchanges scene context while node features retain local geometry. Queries share the initial state and can be evaluated in parallel without recursive feedback. This supports continuous querying within the trained interval, not guaranteed temporal or mesh extrapolation. Architecture and staggered-state alignment are detailed in Appendix B.

Constraint-Preserving Decoder. The decoder adds a bounded deformation residual to a prescribed motion carrier:

$$\begin{aligned} \widetilde{\mathbf{r}}_{ki}(t) &= \mathcal{C}_{ki}(\mathcal{P}, t) + 5t\, g_k(\mathcal{P}) \tanh(\mathbf{u}_{ki}), \\ \widehat{\mathbf{Q}}_{ki}(t) &= \exp(4\pi t[\tanh(\mathbf{q}_{ki})]_\times)\, \mathbf{Q}_{0,ki}. \end{aligned} \tag{5}$$

Both raw outputs use componentwise $\tanh$ before time scaling. The rotation increment is a learned coordinate in the Lie algebra, not a decoded angular velocity. Its exponential preserves $\mathrm{SO}(3)$.

For anchored rods, $\mathcal{C} = \mathbf{r}_0$; for free rods, it provides damped rigid transport fitted to the initial velocity field. A fixed gate selects carrier-only or residual-corrected positions (Appendix B). After decoding, we project free rods outside active support planes and enforce prescribed positions and frames at fixed nodes. These operations preserve feasible initial configurations, but not necessarily their time derivatives. Other contacts are learned rather than enforced by hard non-penetration constraints.

3.4 State and Geometric Supervision

We supervise configurations and their spatial and temporal structure:

$$\begin{aligned}\mathcal{L} = \lambda_{\mathrm{pos}}\mathcal{L}_{\mathrm{pos}} + \lambda_{\mathrm{com}}\mathcal{L}_{\mathrm{com}} + \lambda_{\mathrm{rot}}\mathcal{L}_{\mathrm{rot}} + \lambda_{\mathrm{tan}}\mathcal{L}_{\mathrm{tan}} \\ + \lambda_{\mathrm{len}}\mathcal{L}_{\mathrm{len}} + \lambda_{\mathrm{shear}}\mathcal{L}_{\mathrm{shear}} + \lambda_{\mathrm{bend}}\mathcal{L}_{\mathrm{bend}} + \lambda_{\mathrm{vel}}\mathcal{L}_{\mathrm{vel}}.\end{aligned} \tag{6}$$

Position and center-of-mass terms measure length-normalized errors in local shape and global displacement. Rotation uses the squared geodesic angle between predicted and target frames. Tangent and relative edge-length terms penalize local kinks and stretch; shear/stretch and bend/twist terms compare material geometry derived from Equation 2. Appendix C gives the discrete definitions. No loss requires an additional velocity or strain output head.

Velocity supervision compares neighboring time-query differences with recorded simulator velocities. Padding is masked and losses are averaged per trajectory. Appendix C specifies the temporal stencil, late-time weighting, and distinct pretraining and horizon-adaptation objectives.

3.5 Rendering from Predicted Dynamics

We project predicted configurations with one orthographic map $\pi_C(\mathbf{r}) = S_C A_C \mathbf{r} + \mathbf{b}_C$ throughout each sequence. The viewing axes A_C, scale S_C, and centering $\mathbf{b}_C$ are fixed across frames. Centerline guides join ordered projected nodes, with colors distinguishing rods; they convey shape but not material-frame orientation.

The frame-aware rendering variant additionally projects signed rails:

$$\mathbf{r}^{\pm}_{ki} = \widehat{\mathbf{r}}_{ki} \pm w_k \widehat{\mathbf{d}}_{1,ki}, \qquad w_k = \max(\alpha_{\mathrm{rail}} r^{\mathrm{rad}}_k, w_{\min}). \tag{7}$$

Different colors distinguish the two sides and expose cross-sectional orientation. The display half-width w_k uses radius r^{rad}_k, scale α_{rail}, and minimum extent $w_{\min}$, not a measured ribbon width. Wider strokes produce a corridor mask. The released wrapper uses VACE's default mask handling; protocols using the exported mask must supply it explicitly.

Wan-VACE synthesizes the video from these conditions and the prompt. The 2D guides do not resolve depth ordering or guarantee collision-free video. We assess 3D accuracy separately from video adherence, holding renderer settings fixed when comparing guides.

4 Cosserat-12K Dataset

We construct Cosserat-12K because video corpora lack the distributed states and physical conditions needed to supervise deformation. Its 12,000 trajectories span 24 mechanics templates in six categories: boundary response, external loading, instability, environment contact, self-contact/impact, and multi-body/active motion. Each template fixes rod count, attachments, obstacle topology, and loading form. Its 500 instances independently vary geometry, material, initial state, and controls.

Elastica supplies distributed state labels and physical conditions. In-distribution (ID) tests sample new instances from 18 seen templates; composition out-of-distribution (OOD) testing withholds the fourth template of each family. This tests six new arrangements within the same mechanics taxonomy, not unrestricted topology or object-count transfer. Parameter OOD tests values outside training ranges. Appendix E gives the simulation settings, splits, and parameter ranges.

5 EXPERIMENTS

In this section, we evaluate CosseratGen for physically grounded and controllable video generation. Subsection 5.1 evaluates dynamics prediction and physical parameter control, Subsection 5.2 evaluates operator components, and Subsection 5.3 compares video generation results with baselines and presents qualitative ablations and human evaluation.

5.1 DYNAMICS PREDICTION AND PHYSICAL CONTROL

To evaluate whether DT-NCO captures distributed deformation, we compare predicted centerlines with simulator trajectories on the Cosserat-12K dataset. Static, constant-velocity, and ballistic baselines retain the initial shape or extrapolate motion with constant velocity or gravitational acceleration. MeshGraphNets (Pfaff et al., 2020) predicts dynamics through graph message passing. FNO (Li et al., 2020) learns operators in Fourier space, while RIGNO (Mousavi et al., 2026) uses regional graphs for irregular domains. GAOT (Wen et al., 2026) uses geometry-aware attention, and Transolver++ (Luo et al., 2025) aggregates mesh points into learned physical states for scalable PDE prediction. The DT-NCO predictor, combining a physics-based motion prior with a learned deformation correction, achieves the lowest centerline RMSE across all three splits (Table 1).

Table 1: Centerline prediction errors. ID: in-distribution; Comp./Param. OOD: composition/parameter out-of-distribution tests. Additional evaluation details are in Appendix H.

Method	Centerline RMSE (cm)↓		
	ID	Comp. OOD	Param. OOD
Static	16.46	38.87	16.42
Const. velocity	16.72	39.25	16.72
Ballistic	62.03	41.36	62.25
MeshGraphNets	19.51	39.33	19.43
FNO	3.64	5.79	4.56
RIGNO	3.20	5.72	4.00
GAOT	2.63	4.57	3.58
Transolver++	2.38	4.46	2.95
DT-NCO	**2.16**	**4.35**	**2.67**

Centerline RMSE is 2.16, 4.35, and 2.67 cm on ID, composition OOD, and parameter OOD, respectively. Results on new instances and six held-out arrangements show that DT-NCO predicts centerline deformation beyond its training examples within the same mechanics families. Longer-horizon experiments are described in Appendix H.

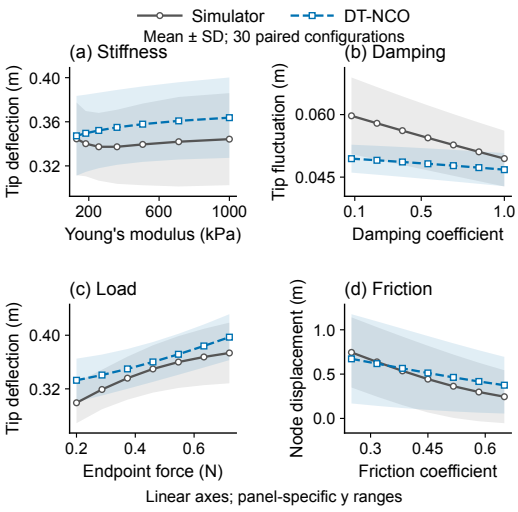

Figure 3: Physical parameter control with DT-NCO. Simulator and predicted mean responses (m) over 30 paired configurations at seven levels; bands show one sample standard deviation. Gray solid and blue dashed curves denote the simulator and DT-NCO, respectively, with one parameter varied per panel. Appendix L defines the metrics.

Parameter Controllability. In Figure 3, we vary one physical input at a time while keeping the remaining configuration fixed. Mean predictions show greater tip deflection with larger loads, weaker oscillation with increased damping, and less displacement with higher friction. However, DT-NCO underestimates damping sensitivity and overestimates displacement at high friction. It does not recover stiffness ordering; the simulator response is itself non-monotonic over this transient window.

Initial Material Frames. To isolate the information in initial orientation, we rotate only $\mathbf{Q}_0$ by $0/30/60/90°$ under gravity, harmonic forcing, tip loading, and compression. Both predictors share centerlines, physical attributes, boundary conditions, training data, and losses. Zeroing unavailable director channels preserves capacity, and orientation groups stay together across splits. Initial velocities and strains are zero, making full-frame and full-state inputs equivalent here (Appendix D).

Full-frame inputs reduce centerline RMSE by 12.68% (Table 2). Thus, initial orientation improves anisotropic bending prediction beyond centerline geometry.

Table 2: Capacity-matched DT-NCO input ablation on 192 held-out ribbon trajectories. Values are mean±sample standard deviation across three seeds. Pos./Norm./Dir.: centerline RMSE (cm), length-normalized RMSE (%), and director error (°). Bold marks the lowest means.

Input	Pos.↓	Norm.↓	Dir.↓
Centerline only	2.61±0.02	4.40±0.04	30.70±0.37
Full frame $\mathbf{Q}_0$	**2.28**±0.01	**3.84**±0.03	**20.58**±0.25
Reduction (%)↑	12.68	12.73	32.98

Inverse Control. Differentiating final-shape error through frozen DT-NCO, we optimize endpoint forces for held-out cantilevers and verify the optimized forces with independent Elastica simulations. Experimental details and results are provided in Appendix H.

5.2 Ablation Studies

We evaluate the roles of physical inputs and output constraints in an separate matched diagnostic (Table 3; protocol in Appendix G). The full model achieves the lowest errors across all four prediction metrics. Removing physical attributes increases RMSE from 2.20 to 8.53 cm; removing rotation or strain supervision also increases prediction errors. Without SO(3) decoding, RMSE rises to 3.30 cm. Rotation supervision also reduces shear/stretch error, while strain supervision improves director accuracy. Both losses therefore benefit quantities beyond their direct supervision targets. These results support the complementary roles of physical conditioning, state supervision, and output constraints in accurate deformation prediction.

Real-World Training. On five cables and ropes from the DEFORM dataset (Chen et al., 2024), we compare real-only training with simulation pretraining followed by fine-tuning under identical real supervision. The quantitative comparison and training protocol are provided in Appendix J. Figure 4 further illustrates real-world motion transfer to two generated appearances. The same predicted guide drives both appearances, keeping the physical motion condition fixed while different prompts change the rendered scene.

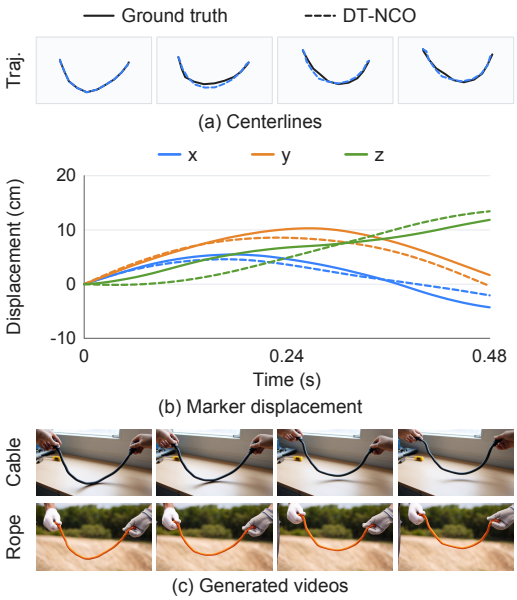

Figure 4: Real-world motion transfer. (a) Centerlines. (b) Central-marker displacements from the initial position, colored by axis. Solid: ground truth; dashed: DT-NCO. (c) Two appearances generated from the same predicted guide.

Table 3: DT-NCO component ablations on an independently reconstructed four-family diagnostic: 192 test trajectories, mean±sample SD over three seeds. Units: RMSE (cm), director (degrees), bend/twist (m^{-1}); shear/stretch is scaled by 10^2. Bold includes all column minima and displayed ties.

Variant	RMSE↓	Director↓	Shear/stretch↓	Bend/twist↓
w/o physical attributes	8.53±0.19	31.35±0.04	12.42±0.30	2.27±0.01
w/o rotation loss	4.75±0.62	29.27±0.40	14.40±0.35	2.16±0.00
w/o strain losses	4.15±0.27	26.09±0.16	12.86±0.10	2.33±0.01
w/o SO(3) decoding	3.30±0.14	27.28±0.39	13.23±0.29	2.22±0.01
w/o boundary projection	4.23±0.37	27.02±0.58	12.29±0.15	2.33±0.03
Full model	**2.20**±0.22	**23.09**±0.30	**12.01**±0.26	**2.09**±0.01

5.3 Video Generation and Control

We compare baseline generators and guidance conditions on 72 shared scenes (Table 4), using released conditioning formats that differ across model families. Metrics are symmetric ROI Chamfer for structural adherence, VBench background consistency and motion smoothness (Huang et al.,

2024), and VideoCon-Physics (VC-Phys) (Bansal et al., 2025). Appendix K provides evaluation details and an additional endpoint-interpolation comparison.

Table 4: Video generation. Sym. dist.: region-of-interest Chamfer (pixels); BC/MS: background consistency/motion smoothness; VC-Phys: physical plausibility. No structure uses the prompt alone. The final two rows compare guidance conditions within frozen VACE. Bold includes displayed ties.

Method / condition	Sym. dist.↓	BC↑	MS↑	VC-Phys↑
General video generators				
CogVideoX-5B	17.010	0.970	0.991	0.431
Wan-T2V-1.3B	19.330	0.962	0.994	0.373
Motion-controlled generators				
Tora	12.020	0.970	0.992	0.436
MagicMotion	2.950	0.917	0.991	0.491
SG-I2V	7.140	0.918	0.973	0.633
Shared frozen VACE generator				
No structure	15.090	0.974	0.994	0.392
Static guide	13.290	0.985	0.996	0.408
Ours	**2.347**	**0.994**	**0.997**	**0.738**

CosseratGen achieves the lowest structural distance and highest physical-plausibility score. Within frozen VACE, evolving guidance reduces structural distance by 82.34%, with the prompt, renderer, and seed fixed. Predicted deformation thus improves control without updating the generator.

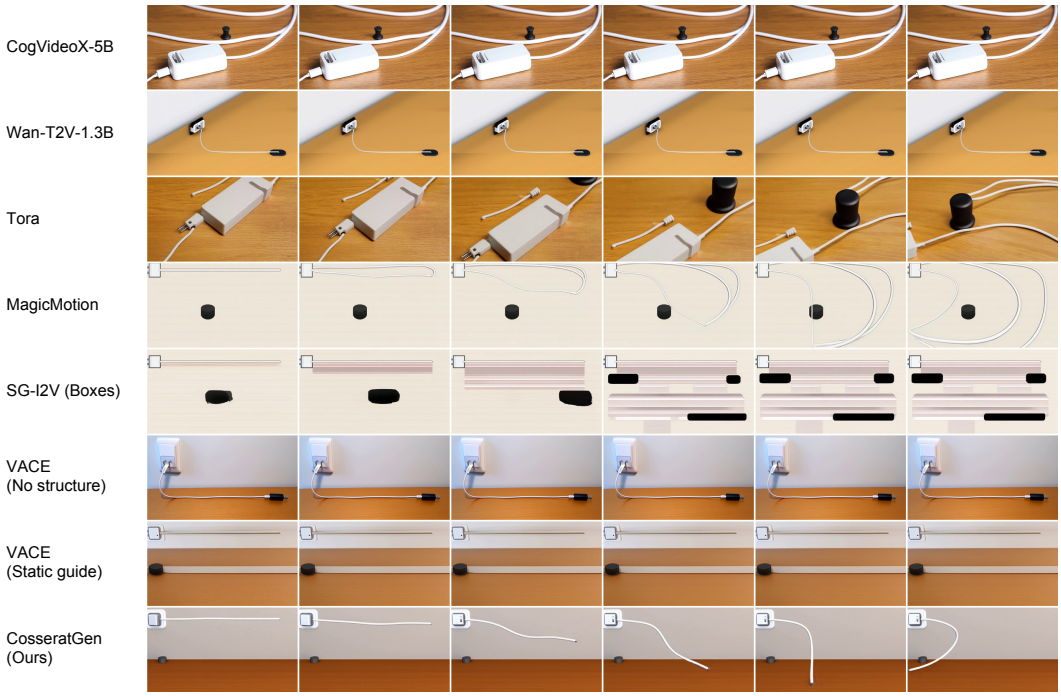

Figure 5: Video generation on a charging cable scene: six ordered frames per method. Text-conditioned methods share the prompt; SG-I2V receives the matched reference and trajectory boxes.

Our generated cable bends and curls while remaining attached; static guidance retains the initial straight shape (Figure 5). Figure 6 shows diverse scenes.

Ablation Studies. In Figure 7, the video guided by the centerline-only predictor bends more sharply downward in the later frames than the simulator-guided result. The video guided by the full-frame predictor more closely follows that bend with the same frozen renderer. Together with Table 2, this illustrates how the gain in centerline prediction translates into more faithful bending guidance, reflected in the generated ribbon motion. Removing physical inputs also reduces the excursion in Figure 8. Additional load- and friction-control experiments are provided in Appendix L.



Figure 6: Video generation across outdoor, industrial, and aquatic scenes. Wheat stalks, rod arrays, cable loops, ribbons, kite tails, and jellyfish tentacles each appear in four ordered frames.

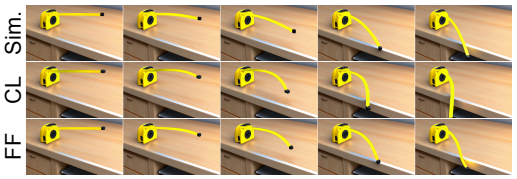

Figure 7: Predictor-input ablation in generated ribbon motion. Sim.: simulator; CL: centerline-only; FF: full-frame. Full-frame and centerline-only predictors use initial states with and without material frames, respectively. With a shared frozen renderer, the full-frame predictor's rollouts reproduce simulator-guided bending more closely.

Human Evaluation. To evaluate perceptual quality, we recruit 40 participants for a user study. Following WonderPlay (Li et al., 2025c), each participant completes ten blinded two-alternative forced-choice (2AFC) trials, judging randomly ordered video pairs on physical plausibility, motion fidelity, and visual quality (Appendix M). Table 5 reports participant-averaged preferences of 66.20–87.50% for CosseratGen across the reported baselines and criteria; the last two rows use frozen VACE. These results demonstrate the advantage of CosseratGen in perceived motion plausibility, naturalness, and visual quality.

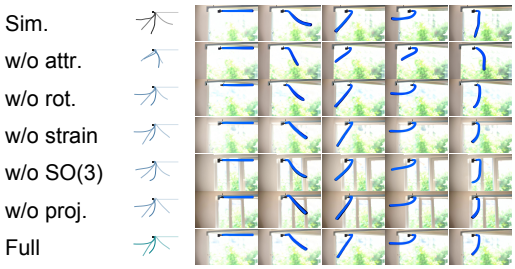

Figure 8: Component ablations in motion (left) and video (right). Only the guide changes across five matched frames. Sim./Full: simulator/full model; w/o: removal of physical attributes (attr.), rotation loss (rot.), strain losses, $\mathrm{SO}(3)$ decoding, or boundary projection (proj.).

Table 5: Human preference for CosseratGen (%). P.P.: physical plausibility; M.F.: motion fidelity; V.Q.: visual quality. Preferences are averaged over participants who evaluated each baseline.

	P.P.↑	**M.F.↑**	**V.Q.↑**
Over CogVideoX-5B	82.30%	85.10%	74.70%
Over Wan-T2V-1.3B	84.20%	87.50%	76.30%
Over Tora	76.00%	79.10%	71.80%
Over MagicMotion	72.30%	78.30%	66.20%
Over SG-I2V	69.10%	73.50%	68.20%
Over no structure	81.60%	84.90%	73.10%
Over static guide	74.10%	80.60%	69.70%

6 Conclusion

We introduce CosseratGen, a controllable video generation framework for slender deformable objects. DT-NCO learns distributed dynamics from simulations to guide a frozen video generator. Experiments demonstrate improved deformation prediction with initial material frames and improved video adherence with evolving guidance. Real-world fine-tuning and differentiable force optimization further demonstrate the value of learned dynamics for connecting physical control with video generation.

Limitations. While CosseratGen enables physically controllable video generation for slender objects, its current formulation focuses on rod-based deformation represented by centerlines and material frames. A natural next step is to extend the framework to shells and solids through corresponding physical representations and supervision. Such extensions could support a broader range of deformable objects while preserving the separation between learned dynamics and appearance synthesis.

7 AI USE, ETHICS, AND REPRODUCIBILITY STATEMENTS

AI Use Statement. Generative AI tools assisted with research planning, manuscript drafting and editing, code development and debugging, experimental workflows, and figure preparation. The use of generative video models within the proposed framework and experiments is described in the corresponding sections. The authors take responsibility for the final content of this work, including text, claims, code, and visual artifacts produced with AI assistance.

Ethics Statement. CosseratGen generates videos with controllable physical motion. Such videos could be mistaken for real recordings and used to spread misinformation. Generated content should therefore be clearly identified as synthetic.

Reproducibility Statement. The main paper describes the model architecture and training objective. The appendix provides simulation settings, data splits, hyperparameters, evaluation metrics, and random seeds, together with the real-trajectory and human-evaluation protocols. Additional qualitative results illustrate the generated motion and the effects of different guidance conditions.

REFERENCES

Stuart S Antman. *Nonlinear problems of elasticity*. Springer, 2005.

Hritik Bansal, Zongyu Lin, Tianyi Xie, Zeshun Zong, Michal Yarom, Yonatan Bitton, Chenfanfu Jiang, Yizhou Sun, Kai-Wei Chang, and Aditya Grover. Videophy: Evaluating physical commonsense for video generation. In *International Conference on Learning Representations*, volume 2025, pp. 102075–102121, 2025.

Miklós Bergou, Max Wardetzky, Stephen Robinson, Basile Audoly, and Eitan Grinspun. Discrete elastic rods. In *ACM siggraph 2008 papers*, pp. 1–12. 2008.

Ryan Burgert, Yuancheng Xu, Wenqi Xian, Oliver Pilarski, Pascal Clausen, Mingming He, Li Ma, Yitong Deng, Lingxiao Li, Mohsen Mousavi, et al. Go-with-the-flow: Motion-controllable video diffusion models using real-time warped noise. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13–23. IEEE, 2025.

Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, and David Duvenaud. Neural ordinary differential equations. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

Yizhou Chen, Yiting Zhang, Zachary Brei, Tiancheng Zhang, Yuzhen Chen, Julie Wu, and Ram Vasudevan. Differentiable discrete elastic rods for real-time modeling of deformable linear objects. *arXiv preprint arXiv:2406.05931*, 2024.

Jimmy Envall, Bernhard Thomaszewski, and Stelian Coros. Understanding the impact of modeling abstractions on motion planning for deformable linear objects. In *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 21741–21748. IEEE, 2025.

Lin Geng Foo, Mark He Huang, Alexandros Lattas, Stylianos Moschoglou, Thabo Beeler, and Christian Theobalt. Physical simulator in-the-loop video generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4301–4311, 2026.

Mattia Gazzola, Levi H Dudte, Andrew G McCormick, and Lakshminarayanan Mahadevan. Forward and inverse problems in the mechanics of soft filaments. *Royal Society open science*, 5(6): 171628, 2018.

Nate Gillman, Charles Herrmann, Michael Freeman, Daksh Aggarwal, Evan Luo, Deqing Sun, and Chen Sun. Force prompting: Video generation models can learn and generalize physics-based control signals. In *Advances in Neural Information Processing Systems*, volume 38, pp. 103174–103201, 2026.

Rohit Girdhar, Mannat Singh, Andrew Brown, Quentin Duval, Samaneh Azadi, Sai Saketh Rambhatla, Akbar Shah, Xi Yin, Devi Parikh, and Ishan Misra. Factorizing text-to-video generation by explicit image conditioning. In *European Conference on Computer Vision*, pp. 205–224. Springer, 2024.

Yuwei Guo, Ceyuan Yang, Anyi Rao, Maneesh Agrawala, Dahua Lin, and Bo Dai. Sparsectrl: Adding sparse controls to text-to-video diffusion models. In *European Conference on Computer Vision*, pp. 330–348. Springer, 2024.

Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P. Kingma, Ben Poole, Mohammad Norouzi, David J. Fleet, and Tim Salimans. Video diffusion models. *Advances in Neural Information Processing Systems*, 35:8633–8646, 2022.

Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, et al. Vbench: Comprehensive benchmark suite for video generative models. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21807–21818. IEEE, 2024.

Zeyinzi Jiang, Zhen Han, Chaojie Mao, Jingfeng Zhang, Yulin Pan, and Yu Liu. Vace: All-in-one video creation and editing. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 17191–17202. IEEE, 2025.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Chenyu Li, Oscar Michel, Xichen Pan, Sainan Liu, Mike Roberts, and Saining Xie. Pisa experiments: Exploring physics post-training for video diffusion models by watching stuff drop. *arXiv preprint arXiv:2503.09595*, 2025a.

Dacheng Li, Yunhao Fang, Yukang Chen, Shuo Yang, Shiyi Cao, Justin Wong, Michael Luo, Xiaolong Wang, Hongxu Yin, Joseph Gonzalez, et al. Worldmodelbench: Judging video generation models as world models. *Advances in Neural Information Processing Systems*, 38, 2026.

Quanhao Li, Zhen Xing, Rui Wang, Hui Zhang, Qi Dai, and Zuxuan Wu. Magicmotion: Controllable video generation with dense-to-sparse trajectory guidance. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 12112–12123. IEEE, 2025b.

Zizhang Li, Hong-Xing Yu, Wei Liu, Yin Yang, Charles Herrmann, Gordon Wetzstein, and Jiajun Wu. Wonderplay: Dynamic 3d scene generation from a single image and actions. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9080–9090. IEEE, 2025c.

Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. *arXiv preprint arXiv:2010.08895*, 2020.

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, 2024.

Huakun Luo, Haixu Wu, Hang Zhou, Lanxiang Xing, Yichen Di, Jianmin Wang, and Mingsheng Long. Transolver++: An accurate neural solver for pdes on million-scale geometries. *arXiv preprint arXiv:2502.02414*, 2025.

Fanqing Meng, Jiaqi Liao, Xinyu Tan, Wenqi Shao, Quanfeng Lu, Kaipeng Zhang, Yu Cheng, Dianqi Li, Yu Qiao, and Ping Luo. Towards world simulator: Crafting physical commonsense-based benchmark for video generation. *arXiv preprint arXiv:2410.05363*, 2024.

Sepehr Mousavi, Shizheng Wen, Levi Lingsch, Maximilian Herde, Bogdan Raonic, and Siddhartha Mishra. Rigno: A graph-based framework for robust and accurate operator learning for pdes on arbitrary domains. *Advances in Neural Information Processing Systems*, 38:150039–150101, 2026.

Koichi Namekata, Sherwin Bahmani, Ziyi Wu, Yash Kant, Igor Gilitschenski, and David Lindell. Sg-i2v: Self-guided trajectory control in image-to-video generation. In *International Conference on Learning Representations*, volume 2025, pp. 38483–38505, 2025.

Sriram Narayanan, Ziyu Jiang, Srinivasa Narasimhan, and Manmohan Chandraker. Phyco: Learning controllable physical priors for generative motion. *arXiv preprint arXiv:2604.28169*, 2026.

Noel Naughton, Jiarui Sun, Arman Tekinalp, Tejaswin Parthasarathy, Girish Chowdhary, and Mattia Gazzola. Elastica: A compliant mechanics environment for soft robotic control. *IEEE Robotics and Automation Letters*, 6(2):3389–3396, 2021.

Karl Pearson. Note on regression and inheritance in the case of two parents. *proceedings of the royal society of London*, 58:240–242, 1895.

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

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.

Siddarth Nilol Kundur Satish, Devesh Jaiswal, Hongyu Chen, and Abhishek Bakshi. Physvideo-generator: Towards physically aware video generation via latent physics guidance. *arXiv preprint arXiv:2601.03665*, 2026.

Han Shao, Tassilo Kugelstadt, Torsten Hädrich, Wojtek Palubicki, Jan Bender, Sören Pirk, and Dominik L Michels. Accurately solving rod dynamics with graph learning. *Advances in Neural Information Processing Systems*, 34:4829–4842, 2021.

Ying Shen, Jerry Xiong, Tianjiao Yu, and Ismini Lourentzou. Phantom: Physics-infused video generation via joint modeling of visual and latent physical dynamics. *arXiv preprint arXiv:2604.08503*, 2026.

Shuwei Shi, Biao Gong, Xi Chen, Dandan Zheng, Shuai Tan, Zizheng Yang, Yuyuan Li, Jingwen He, Kecheng Zheng, Jingdong Chen, et al. Motionstone: Decoupled motion intensity modulation with diffusion transformer for image-to-video generation. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22864–22874. IEEE, 2025.

Charles Spearman. The proof and measurement of association between two things. 1961.

Xiongfeng Tang, Tao Yang, Jian Wang, Guowei Han, Qiran Sun, Hanmin Peng, and Zhijun Sun. Neural operator-based haptic sensing along slender rods. *International Journal of Mechanical Sciences*, pp. 110699, 2025.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

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. In *Advances in Neural Information Processing Systems*, volume 38, pp. 167907–167932, 2026.

Xiang Wang, Hangjie Yuan, Shiwei Zhang, Dayou Chen, Jiuniu Wang, Yingya Zhang, Yujun Shen, Deli Zhao, and Jingren Zhou. Videocomposer: Compositional video synthesis with motion controllability. *Advances in Neural Information Processing Systems*, 36:7594–7611, 2023.

Shizheng Wen, Arsh Kumbhat, Levi Lingsch, Sepehr Mousavi, Yizhou Zhao, Praveen Chandrashekar, and Siddhartha Mishra. Geometry aware operator transformer as an efficient and accurate neural surrogate for pdes on arbitrary domains. *Advances in Neural Information Processing Systems*, 38:155423–155501, 2026.

Bing Wu, Chang Zou, Changlin Li, Duojun Huang, Fang Yang, Hao Tan, et al. Hunyuanvideo 1.5 technical report. *arXiv preprint arXiv:2511.18870*, 2025.

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.

Jinbo Xing, Menghan Xia, Yong Zhang, Haoxin Chen, Wangbo Yu, Hanyuan Liu, Gongye Liu, Xintao Wang, Ying Shan, and Tien-Tsin Wong. Dynamicrafter: Animating open-domain images with video diffusion priors. In *European Conference on Computer Vision*, pp. 399–417. Springer, 2024.

Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. In *International Conference on Learning Representations*, volume 2025, pp. 83048–83077, 2025.

Guy Yariv, Yuval Kirstain, Amit Zohar, Shelly Sheynin, Yaniv Taigman, Yossi Adi, Sagie Benaim, and Adam Polyak. Through-the-mask: Mask-based motion trajectories for image-to-video generation. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18198–18208. IEEE, 2025.

Yu Yuan, Xijun Wang, Tharindu Wickremasinghe, Zeeshan Nadir, Bole Ma, and Stanley Chan. Newtongen: Physics-consistent and controllable text-to-video generation via neural newtonian dynamics. In *International Conference on Learning Representations*, volume 2026, pp. 12847–12878, 2026.

Haoze Zhang, Tianyu Huang, Zichen Wan, Xiaowei Jin, Hongzhi Zhang, Hui Li, and Wangmeng Zuo. Physchoreo: Physics-controllable video generation with part-aware semantic grounding. *arXiv preprint arXiv:2511.20562*, 2025a.

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.

Zhenghao Zhang, Junchao Liao, Menghao Li, Zuozhuo Dai, Bingxue Qiu, Siyu Zhu, Long Qin, and Weizhi Wang. Tora: Trajectory-oriented diffusion transformer for video generation. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2063–2073. IEEE, 2025b.

Wenliang Zhao, Lujia Bai, Yongming Rao, Jie Zhou, and Jiwen Lu. Unipc: A unified predictor-corrector framework for fast sampling of diffusion models. *Advances in Neural Information Processing Systems*, 36:49842–49869, 2023.

Appendix

A Appendix Introduction

This appendix provides additional discussions and details on the Cosserat state representation and operator design (Section B), loss definitions and material-frame conditioning (Sections C–D), dataset construction and training protocols (Sections E–G), dynamics prediction and real-trajectory evaluation (Sections H–J), video evaluation, physical control, and human evaluation (Sections K–M), physical consistency (Section N), and more visual results (Section O).

B Cosserat State and Operator Details

Let s denote reference arc length, $\mathbf{r}(s,t) \in \mathbb{R}^3$ the centerline, and $\mathbf{Q} = [\mathbf{d}_1, \mathbf{d}_2, \mathbf{d}_3]^\top \in \mathrm{SO}(3)$ the world-to-material rotation. With the conventions of Equation 3, the balance laws are

$$\begin{aligned} \partial_t(\rho A \mathbf{v}) &= \partial_s \mathbf{n} + \mathbf{f}, \qquad \partial_t \mathbf{r} = \mathbf{v}, \\ \partial_t \boldsymbol{\ell} &= \partial_s \mathbf{m} + (\partial_s \mathbf{r}) \times \mathbf{n} + \boldsymbol{\tau}. \end{aligned} \tag{8}$$

Here $\mathbf{n}, \mathbf{m}, \mathbf{f}, \boldsymbol{\tau}$, and $\boldsymbol{\ell}$ are spatial force, moment, distributed force, distributed torque, and angular momentum per reference length. If $\mathbf{J}_m$ is material rotational inertia per reference length (including density), then $\boldsymbol{\ell} = \mathbf{Q}^\top \mathbf{J}_m \boldsymbol{\omega}$, with $\boldsymbol{\omega} = \mathrm{ax}(\mathbf{Q}\partial_t \mathbf{Q}^\top)$. Constitutive maps connect material strains to internal response. Elastica (Naughton et al., 2021) discretizes these equations and handles gravity, damping, constraints, and penalty contact. Simulator solutions supervise state prediction through the losses defined in Section C. Our diagnostics measure simulator-state accuracy and geometric validity.

The initial Cosserat state and complete mechanics specification are

$$\mathcal{S}_0 = (\mathbf{r}_0, \mathbf{v}_0, \mathbf{Q}_0, \boldsymbol{\omega}_0, \boldsymbol{\sigma}_0, \boldsymbol{\kappa}_0), \qquad \mathcal{P} = (\mathcal{S}_0, \mathbf{a}),$$
$$\mathbf{a} = \{\boldsymbol{\mu}, \mathcal{B}, \mathcal{E}, \mathbf{c}, \mathbf{h}\}.$$

Here $\boldsymbol{\mu}$ contains length, radius, density, Young's modulus, and damping. The node-level mask $\mathcal{B}$ identifies fixed nodes, and $\mathcal{E}$ specifies gravity and rigid contact geometry. The four-dimensional $\mathbf{c}$ stores prescribed forces or geometric controls. The 24-dimensional descriptor $\mathbf{h}$ records rod count, clamp pattern, gravity, plane/obstacle contact, friction, self/inter-rod contact, load type, and load magnitude. Scalar and binary fields are concatenated with a load-type code, with inactive fields set to zero.

The encoder scales radius by 50 and density by $1/1000$, and maps Young's modulus to $\log_{10}(\max(E,1))/6$; other attributes are unchanged. Inputs exclude scenario identifiers; the physical-attribute ablation removes these groups jointly.

Architecture and input layout. The node encoder has two width-192 linear layers with SiLU activations. The decoder receives the node code, pooled scene code, and time basis; it has two width-192 hidden layers followed by six outputs. The full model contains 171,410 parameters, of which 170,082 participate in the forward pass. The core used in the frame-input diagnostic has 165,686 parameters.

Each node input concatenates initial position and velocity, local tangent and second spatial difference, the vectorized material frame, angular velocity and strains, physical attributes, rod/material coordinates, and the boundary indicator.

The encoder reserves 18 state channels: nine for $\mathbf{Q}_0$ and three each for angular velocity, shear/stretch strain, and bend/twist strain. All variants retain position and linear velocity. Centerline-only input zeros these channels; $\mathbf{d}_3$-only retains the third director; full-frame retains all nine frame entries; full-state also retains angular velocity and strains. Zeroing channels keeps capacity matched within each architecture.

The default discretization has 20 elements and 21 nodes. Positions and velocities are nodal, frames and angular velocities are element-centered, and curvature lies on Voronoi cells. In the reconstructed diagnostic, element entry e maps to node e; the final element and Voronoi entries are repeated to

obtain 21-node inputs, without rotation interpolation. Strains are recomputed before padding. This alignment is specific to the reconstruction.

Mean pooling over valid nodes supplies scene context. The decoder uses the 12-dimensional monotone time basis, in elapsed seconds,

$$\gamma(t) = [t, t^2, t^3, \sqrt{t+\epsilon} - \sqrt{\epsilon}, \log(1+t), t/(1+t), \\ 1 - e^{-ct}, \tanh(ct)]_{c\in\{1/2,1,2\}}.$$

The decoder predicts bounded displacement and axis-angle increments. Exponential-map decoding preserves $\mathrm{SO}(3)$, and fixed nodes are reset to their initial states. Padding is masked in pooling and losses; batched time queries produce rollouts without autoregressive feedback.

The factorized branches map 4 structural, 7 environmental, and 12 loading attributes through width-24 hidden layers into 16-dimensional outputs; these are summed and passed through SiLU. The geometry input concatenates six raw plane coordinates with a 12-dimensional obstacle code. A shared $5 \to 24 \to 12$ encoder processes each cylinder's relative center, radius, and active flag. Active obstacles receive weights $\mathrm{softmax}_a(-20\|\Delta\mathbf{r}_{a,xy}\|)$ at each node; the aggregate is zero when no obstacle is active.

Carrier and fixed residual gate. Let $f_k = 1$ indicate that rod k has no fixed node. Write $\mathbf{c}_k$ and $\bar{\mathbf{v}}_k$ for masked nodal means, and $\mathbf{x}_{ki} = \mathbf{r}_{0,ki} - \mathbf{c}_k$. A least-squares rigid angular velocity is fitted from the initial translational velocity field:

$$\mathbf{\Omega}_k = \left[\sum_i m_{ki}(\|\mathbf{x}_{ki}\|^2\mathbf{I} - \mathbf{x}_{ki}\mathbf{x}_{ki}^\top)\right]^+ \sum_i m_{ki}\,\mathbf{x}_{ki} \times (\mathbf{v}_{0,ki} - \bar{\mathbf{v}}_k). \tag{9}$$

The pseudoinverse uses relative cutoff 10^{-5}. Define $\beta_k = \max(\mu_{\mathrm{damping},k}, 10^{-4})$, $a_k(t) = (1 - e^{-\beta_k t})/\beta_k$, and $b_k(t) = (t - a_k(t))/\beta_k$. For gravity $\mathbf{g}_{\mathrm{grav}} = (0, \mathcal{E}_0, 0)^\top$, the carrier is

$$\mathcal{C}_{ki}(t) = \begin{cases} \mathbf{c}_k + a_k\bar{\mathbf{v}}_k + b_k\mathbf{g}_{\mathrm{grav}} + \exp(a_k[\mathbf{\Omega}_k]_\times)\mathbf{x}_{ki}, & f_k = 1, \\ \mathbf{r}_{0,ki}, & f_k = 0. \end{cases} \tag{10}$$

The carrier describes damped rigid transport of positions, while directors follow the independent learned update in Equation 5. It does not solve general Euler rotational dynamics or transport directors. With active-plane flag a_{pl} and the encoded unit normal $\mathbf{n}_p$, the exact residual gate is

$$g_k = 1 - f_k\,\mathbf{1}\{a_{\mathrm{pl}} = 0 \text{ or } |(\mathbf{n}_p)_y| < 0.5\}. \tag{11}$$

When $g_k = 0$, positions follow the prescribed motion carrier and any applicable geometric projection; otherwise, the network adds a learned deformation correction. The gate uses geometry and fixed-node attributes, not template identity.

For plane origin $\mathbf{o}_p$ and rod radius r_k^{rad}, the free-node projection after adding the residual is

$$\widehat{\mathbf{r}}_{ki} = \widetilde{\mathbf{r}}_{ki} + f_k a_{\mathrm{pl}}\big[r_k^{\mathrm{rad}} - (\widetilde{\mathbf{r}}_{ki} - \mathbf{o}_p)^\top\mathbf{n}_p\big]_+\mathbf{n}_p. \tag{12}$$

The implementation assumes the supplied normal is unit length. Fixed positions and directors are replaced last. Initial-state exactness requires the initial free nodes to satisfy the plane constraint; the time derivatives of the outputs are not constrained to match the supplied initial velocities exactly.

C Loss Definitions

The pretraining objective is

$$\mathcal{L} = \mathcal{L}_{\mathrm{pos}} + 0.2\mathcal{L}_{\mathrm{com}} + 0.1\mathcal{L}_{\mathrm{rot}} + 0.2\mathcal{L}_{\mathrm{tan}} + 0.05\mathcal{L}_{\mathrm{len}} \\ + 0.01\mathcal{L}_{\mathrm{shear}} + 0.005\mathcal{L}_{\mathrm{bend}} + 0.02\mathcal{L}_{\mathrm{vel}}. \tag{13}$$

The 1.6-second adaptation retains position, rotation, tangent, length, and velocity with weights $1/0.1/0.05/0.05/0.02$ and sets the remaining weights to zero.

Table 6: Centerline RMSE (cm) by family in the capacity-matched DT-NCO ribbon diagnostic. Bold marks the lower error; the last row gives its relative reduction.

Input	Gravity	Harmonic	Tip load	Compression
Centerline only	1.81	4.38	2.46	1.80
Full frame $\mathbf{Q}_0$	**1.74**	**4.04**	**1.77**	**1.58**
Reduction (%)↑	3.90	7.70	28.00	12.60

Position and per-rod center-of-mass errors are normalized by physical rod length. Rotation loss is the squared $\mathrm{SO}(3)$ geodesic angle rather than entrywise matrix error:

$$d_{\mathrm{SO}(3)}(\widehat{\mathbf{Q}}, \mathbf{Q})^2 = \arccos^2\left(\mathrm{clip}\left(\frac{\mathrm{tr}(\widehat{\mathbf{Q}}\mathbf{Q}^\top) - 1}{2}, -1, 1\right)\right). \tag{14}$$

Tangent loss is $1 - \widehat{\mathbf{t}}^\top \mathbf{t}$, and length loss is a robust relative edge-length penalty. The strain terms further relate local geometry to material orientation. For element e with reference length ℓ_e, the shear/stretch strain is

$$\boldsymbol{\sigma}_e = \mathbf{Q}_e(\mathbf{r}_{e+1} - \mathbf{r}_e)/\ell_e - \mathbf{e}_3.$$

The offset $-\mathbf{e}_3$ gives zero strain for an undeformed straight element. Adjacent element material frames yield curvature through the $\mathrm{SO}(3)$ logarithm divided by Voronoi length. For the row-director convention, a discrete physical curvature is $\boldsymbol{\kappa}_e = -\log(\mathbf{Q}_{e+1}\mathbf{Q}_e^\top)^\vee/\bar{\ell}_e$, with $\bar{\ell}_e = (\ell_e + \ell_{e+1})/2$. The reconstructed component diagnostic stores and supervises the opposite signed quantity, $\boldsymbol{\kappa}_e^{\mathrm{diag}} = -\boldsymbol{\kappa}_e$, in both its initial channels and targets. Reversing both signs leaves the symmetric strain-difference penalty unchanged; checkpoint inputs must nevertheless retain their training convention.

Smooth-L_1 strain penalties reduce the influence of isolated high-curvature samples. Velocity consistency compares differences of neighboring predictions with recorded simulator velocities, not differences of target positions. In the reconstructed component diagnostic, an interior frame index $j \in \{1, \ldots, 47\}$ is sampled and the model is evaluated at $t_j - \Delta t, t_j, t_j + \Delta t$, where $\Delta t = 0.8/48$ seconds. This interior stencil is specific to the diagnostic. Later frames receive weight $w(t) = 1/3 + 2(t/T)^2$, with unit continuous mean over $[0, T]$. Padding is masked. Diagnostic position, rotation, and velocity terms exclude fixed nodes; center-of-mass terms use all valid nodes, and geometric terms use valid edges.

Center-of-mass, geometric, and director losses supervise transport, local shape, and cross-sectional orientation, respectively. Losses are averaged within each trajectory and then across the batch, giving trajectories equal weight.

D Paired Wide-Ribbon Frame Diagnostic

To isolate the effect of material-frame input beyond the circular rods in the core benchmark, we construct 2,000 anisotropic wide-ribbon trajectories across gravity, harmonic forcing, tip loading, and compression. Each family contains 125 groups of four, sharing initial centerline, material parameters, boundaries, and world-frame load while varying cross-sectional orientation by $0/30/60/90°$. The bending-stiffness ratio is 400, corresponding to a width-to-thickness ratio of 20. This dataset is independent of Cosserat-12K and its OOD and long-horizon sets.

The held-out test set contains 192 trajectories in 48 complete orientation groups, with no group shared with training or validation. Centerline-only and full-$\mathbf{Q}_0$ variants share 165,686 parameters, data, losses, 400 training epochs, and seeds 17/41/73; excluded state channels are zeroed. Zero initial velocity and strain make full-frame and full-state inputs equivalent here. The general benchmark retains full-state input.

The main paper reports the aggregate result, while Table 6 gives the per-family breakdown. The paired intervention changes only initial cross-sectional orientation; it supports the role of material-frame input for anisotropic objects.

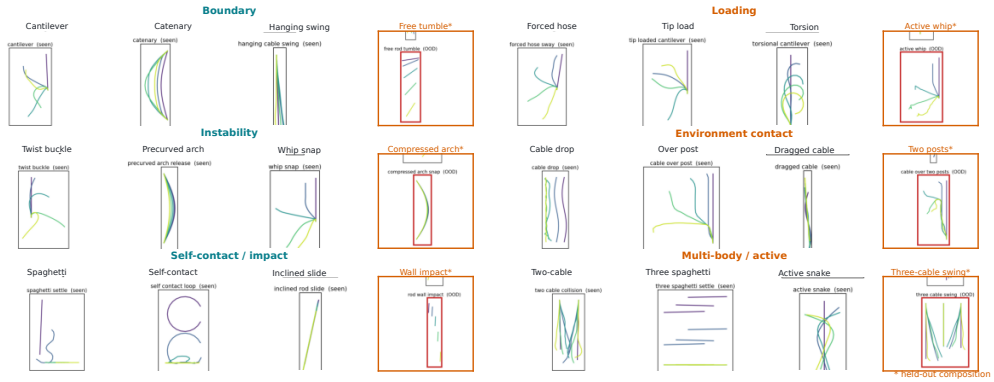

Figure 10: The 24 scenario templates in Cosserat-12K, organized into six mechanics families. Each template contains 500 independently randomized scene instances. Every family has three seen templates and one composition withheld in full, marked by an asterisk.

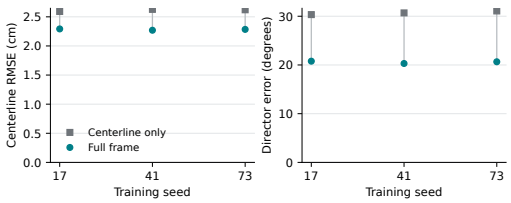

Figure 9: Effect of material-frame input across training seeds in the paired wide-ribbon diagnostic. Each point reports mean centerline RMSE (left, cm) or director error (right, degrees) on the same 192 test trajectories. Connected points pair capacity-matched models trained with the same seed; both axes start at zero.

Figure 9 shows lower centerline and director errors with full-frame conditioning across all three seeds, supporting its role in subsequent shape prediction.

E Dataset Construction and Splits

Elastica integrates 20-element rods at a 10^{-4} s internal step and stores 49 states over 0.8 s, recording centerlines, directors, velocities, strains, and physical conditions for up to three rods. Three templates per category provide 7,200 training, 900 validation, and 900 ID test trajectories; the fourth supplies 3,000 composition-OOD trajectories. Numerical screening rejects non-finite states, coordinates beyond 20 m, and edges longer than twice their rest length. Contact penetration and clamp drift do not filter the accepted set.

Figure 10 identifies all 24 templates and the six fully withheld compositions. Table 7 gives their sampling ranges.

Figure 11 illustrates motion variation from randomized material, geometry, initial-state, and control values. Examples are selected independently of model performance; sampling ranges appear in Table 7.

Parameter-OOD trajectories preserve template geometry but draw density from 562.5–675 or 1430–1625 kg/m^3 and damping from 0.05–0.12 or 0.88–1.05. For an ID log-stiffness range from a to b, with width $w = b - a$, the two disjoint OOD ranges extend from $a - 0.35w$ to $a - 0.15w$ and from $b + 0.15w$ to $b + 0.35w$. Prescribed loads are multiplied by 1.5.

Each template fixes rod count, attachments, contact topology, and loading form; its 500 instances vary continuous physical parameters. ID tests use new instances of seen templates, parameter OOD

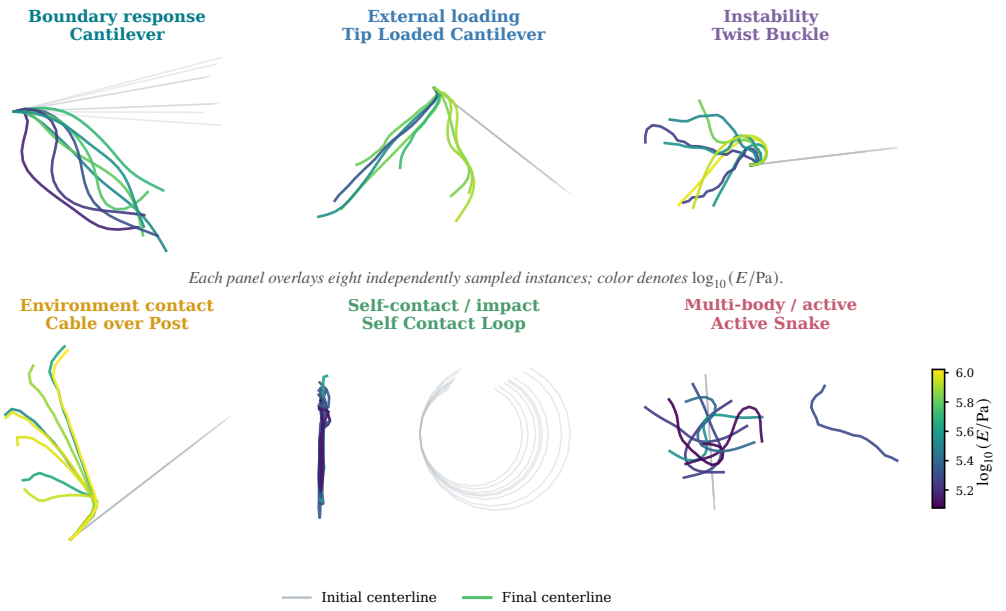

Figure 11: Within-template variation across six mechanics families. Each panel overlays eight instances at fixed, evenly spaced sample indices. Gray and colored curves show initial and final centerlines; color encodes $\log_{10}(E/\mathrm{Pa})$. A joint principal component analysis (PCA) projection and common scale are used within each panel.

Table 7: Sampling ranges for the 24 Cosserat-12K templates. Length L is in meters, radius r in millimeters, and stiffness is $\log_{10}(E/\mathrm{Pa})$. Daggers mark fully held-out composition-OOD templates. Unless overridden below, density is 750–1300 kg/m^3 and damping is 0.15–0.80. Controls marked $\pm$ sample both directions.

Family	Template	L	r	$\log_{10} E$	Principal randomized control or contact variable
Boundary	Cantilever	.28–.32	4–7	5.0–5.8	initial tip velocity $\pm$0.8 m/s
	Catenary	2.30–4.60	4–8	4.8–5.6	sag .08–.16L; force .3–1.0 N at .8–1.8 Hz
	Hanging cable swing	.60–1.50	6–12	4.8–5.7	initial transverse velocity .45–1.05 m/s
	Free-rod tumble†	.45–.90	6–12	5.2–6.1	angular speed 1.0–2.5 rad/s
Loading	Forced hose sway	.60–1.20	9–16	5.3–6.3	force .4–1.5 N at 1.2–2.5 Hz
	Tip-loaded cantilever	.35–.75	5–10	5.1–6.1	endpoint force .15–.75 N
	Torsional cantilever	.30–.60	4–8	5.2–6.2	endpoint torque .001–.006 N m
	Active whip†	.70–1.20	4–9	4.9–5.8	wave amplitude .03–.12 N; 1.0–2.5 Hz
Instability	Twist buckle	.28–.45	4–8	5.2–6.0	torque .001–.006 N m; compression .05–.30 N
	Precurved arch release	.45–.80	4–9	5.3–6.2	span .72–.84L; kick $\pm$.35 m/s
	Whip snap	.75–1.25	4–8	5.0–5.9	initial kick $\pm$.8–1.8 m/s
	Compressed arch snap†	.45–.80	4–9	5.3–6.2	span .58–.72L
Environment contact	Cable drop	1.00–2.00	5–10	4.7–5.5	friction .20–.80
	Cable over post	2.00–4.00	9–16	5.4–6.2	friction .20–.70; post radius .09–.15 m
	Dragged cable	2.00–4.00	7–13	4.9–5.7	friction .25–.65; pull 3–5 N
	Cable over two posts†	2.20–3.60	8–14	5.3–6.1	friction .20–.65; post radius .08–.13 m
Self-contact / impact	Spaghetti settle	.24–.27	2–3.5	5.0–5.7	friction .05–.35
	Self-contact loop	.55–.90	4–8	4.9–5.7	friction .15–.45
	Inclined rod slide	.45–.90	6–12	5.3–6.2	slope .20–.45 rad; friction .05–.35
	Rod wall impact†	.45–.80	6–12	5.2–6.1	impact speed .7–1.4 m/s; friction .10–.40
Multi-body / active	Two-cable collision	.80–1.50	5–10	4.8–5.6	inward speed 1.15–1.55 m/s
	Three-spaghetti settle	.24–.27	2–3.5	4.8–5.5	friction .10–.40
	Active snake	.55–.65	14–22	5.0–5.8	actuation 1–3 N at 1–2 Hz; friction .30–.80
	Three-cable swing†	.70–1.10	5–9	4.9–5.7	initial speed .90–1.35 m/s

moves selected values beyond training ranges, and composition OOD withholds one entire template per family. The long-horizon set extends the query duration. All core splits use 20 elements per

rod and at most three rods; these tests measure mechanics generalization, while simulator-guided galleries illustrate visual diversity.

F Training and Reproducibility

Pretraining uses 200 epochs, learning rate 2×10^{-3}, cosine decay, weight decay 10^{-5}, batch size 256, and gradient clipping at norm 2. A further 60 epochs at 3×10^{-4} add 540 training trajectories over 1.6 s, each sampled twice per epoch. Validation and test trajectories are disjoint; the 480-trajectory test set includes 180 seen-template and 300 unseen-template instances. The adaptation objective is given in Section C; checkpoints are selected by validation error.

Fine-tuning balances physical attributes without template labels or composition-OOD training data. MeshGraphNets provides an autoregressive reference.

Table 8 measures mechanics-stage latency after warm-up with preloaded inputs and synchronized CUDA timing, excluding rendering. DT-NCO evaluates all 49 query times jointly.

Table 8: Mechanics-stage latency for a 49-state rollout on one RTX 4080 SUPER. Median and 95th-percentile (P95) times are measured after warm-up with preloaded inputs and checkpoints, excluding rendering. Bold marks lower latency.

Method	Execution	Median (ms)↓	P95 (ms)↓
MeshGraphNets	49 autoregressive steps	94.02	101.45
DT-NCO	**49 parallel queries**	**4.13**	**4.83**

Timing uses 60 repeats for DT-NCO and 30 for MeshGraphNets. An independent 80-repeat experiment gives 4.00–4.21 ms for 1–97 batched query times, showing little latency growth over this range on the tested device.

G Component Diagnostic Protocol

Purpose and data. Table 3 tests the roles of physical conditioning, orientation supervision, strain supervision, and output constraints in an independently reconstructed dataset, separate from the original benchmark and the cross-sectional-orientation study. We generate four PyElastica families: gravity-driven bending, constant tip loading, harmonic forcing, and release from a curved initial configuration. Each physical parameter group contains four initial cross-sectional orientations, $0°$, $30°$, $60°$, and $90°$, with identical initial centerlines. Splits are assigned by parameter group, so orientation variants cannot cross split boundaries. After excluding three nonfinite groups, the dataset contains 1,588 training, 192 validation, and 192 test trajectories, with 16 unused guard trajectories.

Each example contains one anchored strip discretized into 20 elements and 21 nodes, with 49 states over 0.8 seconds. Length varies from 0.35 to 0.60 m; width and thickness are fixed at 0.025 and 0.00125 m. The Young's modulus is sampled logarithmically between $10^{6.7}$ and $10^{7.3}$ Pa, density is $1{,}000$ kg/m^3, and the damping parameter ranges from 0.5 to 2.0. Applied force magnitudes range from 0.025 to 0.12 N, and harmonic frequencies from 1.0 to 2.5 Hz. Gravity is 9.81 m/s^2. This diagnostic excludes inter-object contact. Initial strains use the same discrete convention as the training losses.

Matched training. All variants share data, full-state inputs, query sampling, and seeds 17/41/73. Training uses 400 epochs of AdamW, batch size 64, learning rate 3×10^{-4} with cosine decay to 3×10^{-5}, weight decay 10^{-5}, and gradient clipping at 1. The final checkpoint is evaluated without selection. The full model has 171,410 parameters; unconstrained matrix output increases this to 172,568.

Ablation definitions. The full model uses the loss weights in Equation 13. The rotation-loss ablation removes only direct orientation supervision; orientation information remains in the strain losses. The strain-loss ablation removes both shear/stretch and bend/twist terms while retaining the

tangent and edge-length losses. The physical-attribute ablation zeros the material, control, environment, physics, and geometry descriptors while retaining initial state and boundary masks. In this anchored, contact-free setting, the carrier is independent of the removed inputs. The remaining variants remove the constrained rotation decoder or hard boundary projection, respectively. All other settings are unchanged.

Metric interpretation. Metrics are first averaged within each test trajectory and then across the fixed test set. Centerline RMSE is the square root of mean squared Euclidean position error over all nodes and frames. Director error is the mean geodesic angle. For every variant, Gram–Schmidt orthonormalization of the predicted frame is used in orientation and strain comparisons; the unconstrained variant's raw matrix is retained when computing $\|\widehat{\mathbf{Q}}\widehat{\mathbf{Q}}^{\mathsf{T}} - \mathbf{I}\|_F$. Strain metrics report componentwise mean absolute error in shear/stretch and bend/twist, with the latter measured in m^{-1}. Clamp drift is the maximum fixed-node displacement over time, averaged across test trajectories. Tables report the mean and sample standard deviation across the three training seeds.

Qualitative coverage. Figures 12, 13, 14, and 15 show the same group and orientation chosen before training in each family, using seed 17 for every variant. All panels share an orthographic projection and scale; quantitative errors are computed in three dimensions.

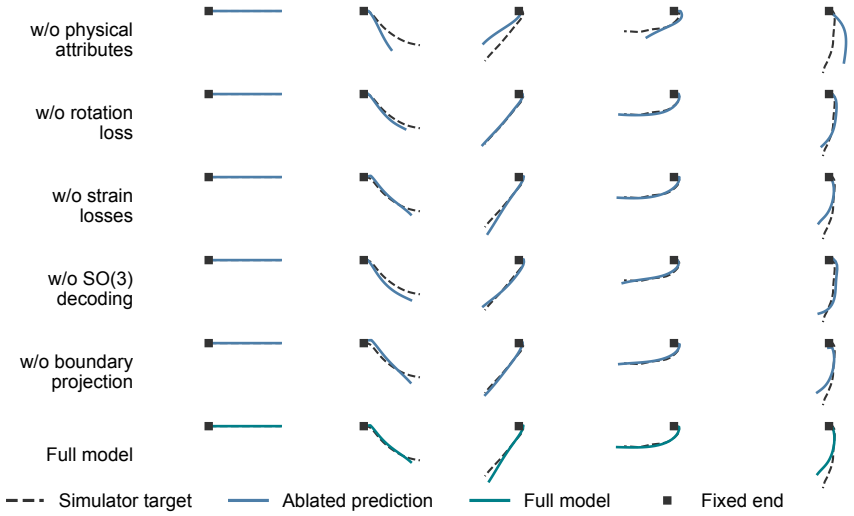

Figure 12: Component ablations under gravity-driven bending. Rows follow Table 3; columns show five query times. Black: simulator; colored: prediction; squares: fixed endpoints.

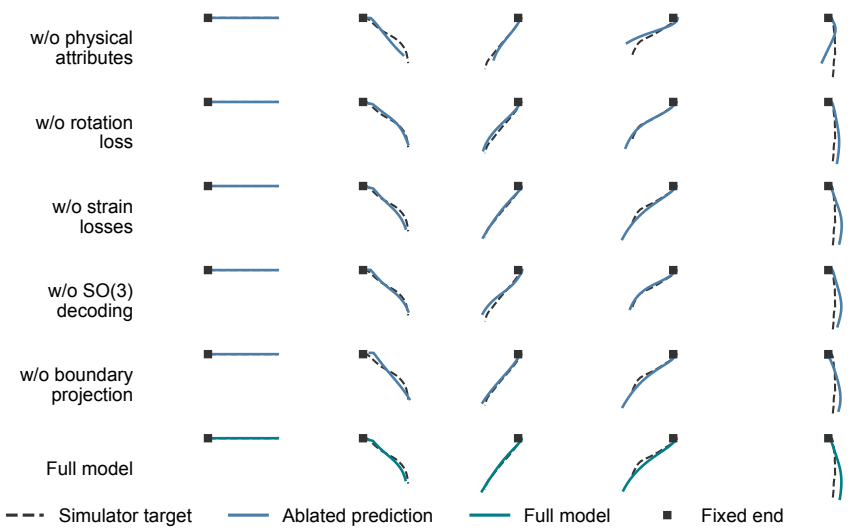

Figure 13: Component ablations under tip loading. Rows follow Table 3; columns show five query times. Black: simulator; colored: prediction; squares: fixed endpoints.

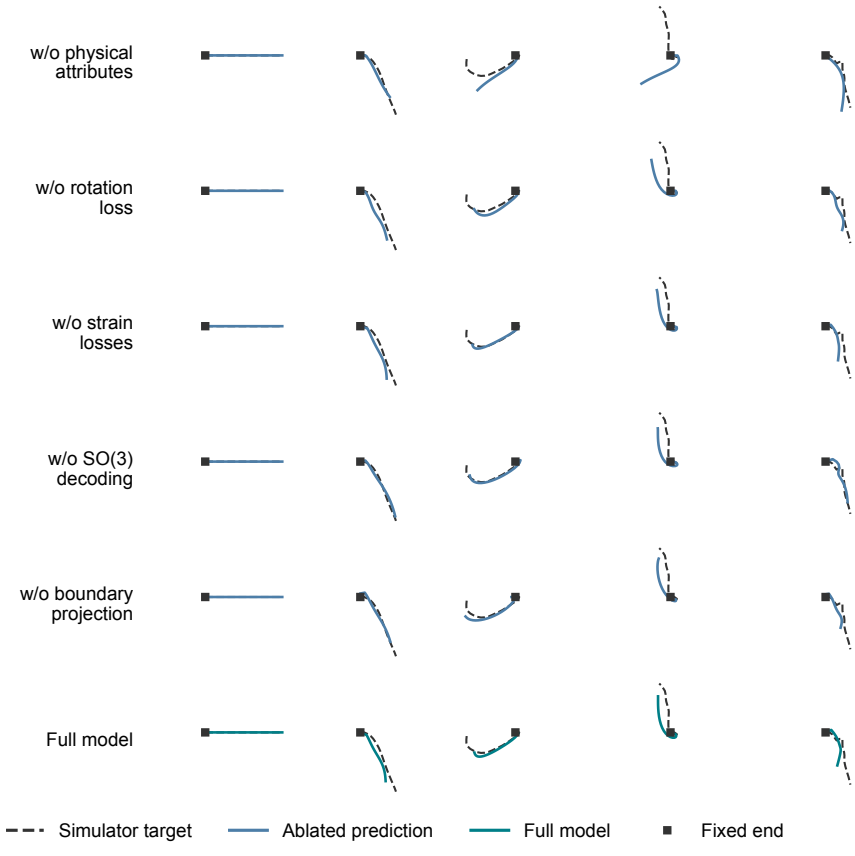

Figure 14: Component ablations under harmonic forcing. Rows follow Table 3; columns show five query times. Black: simulator; colored: prediction; squares: fixed endpoints.

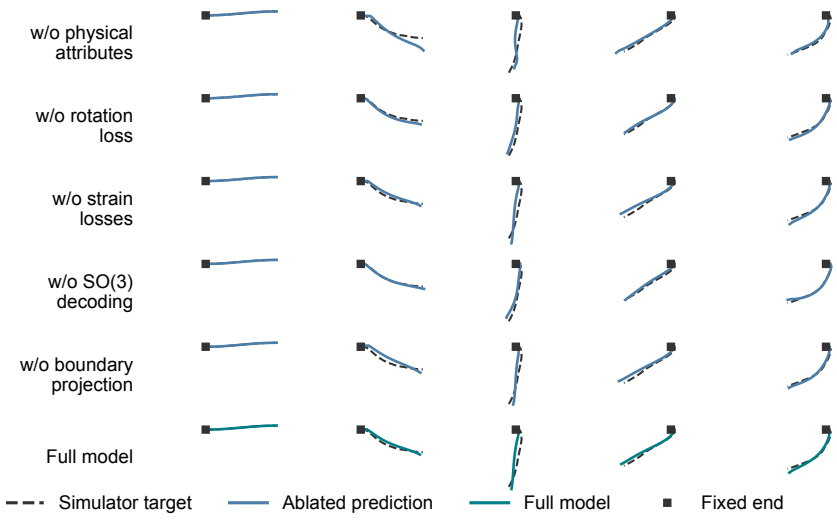

Figure 15: Component ablations under curved release. Rows follow Table 3; columns show five query times. Black: simulator; colored: prediction; squares: fixed endpoints.

Table 9: Effect of training horizon on centerline prediction. Both methods use the same data splits, 10,000 training updates, and seed 17. Values are mean per-trajectory RMSE (cm) on 176 test trajectories. Bold marks the lower error at each test horizon within each training-horizon group.

		Test horizon: RMSE (cm)↓		
Training horizon	Method	0.8 s	1.2 s	1.6 s
0.8 s	MeshGraphNets	5.44	13.87	48.17
	DT-NCO	**2.08**	**5.69**	**15.11**
1.6 s	MeshGraphNets	7.32	12.92	20.23
	DT-NCO	**3.57**	**3.63**	**3.82**

H Extended Dynamics Results

Table 9 compares direct-time prediction and autoregressive rollout under matched training horizons on the reconstructed diagnostic dataset.

The following analyses examine template-level errors and differentiable endpoint-force control. Position error is averaged first over valid material points, then over trajectories, giving each trajectory equal weight.

H.1 Temporal Coverage

We separate temporal extrapolation from prediction within the training horizon. The reconstructed benchmark contains 1,488 training, 184 validation, and 176 test trajectories across gravity, tip loading, harmonic forcing, and curved release. Physical parameter groups are disjoint across splits, with all four initial frame orientations kept together. Both methods are trained from scratch for 10,000 updates with batch size 64, using either the first 0.8 s or the full 1.6 s of each training trajectory. MeshGraphNets uses the NVIDIA implementation with a Cosserat task adapter and autoregressive steps of $1/60$ s. Table 9 reports the prespecified representative configurations with seed 17; the two methods retain their respective training losses. This reconstructed comparison is separate from the main-paper Cosserat-12K benchmark.

Table 9 shows lower error for DT-NCO at both training horizons. Extending training coverage improves long-horizon predictions for both methods; the 1.6-s training condition evaluates within-horizon prediction.

H.2 Position Accuracy and Material-Frame Accuracy

Figure 16 compares predicted and simulated centerlines on ID and held-out compositions. Centerline error, director error, and $\mathrm{SO}(3)$ validity assess shape, orientation, and rotation validity, respectively. The ribbon experiment (Table 2) tests initial-frame conditioning. Centerline guides use $\widehat{\mathbf{r}}$; signed rails additionally use $\widehat{\mathbf{d}}_1$, whose error is not isolated in the video tests.

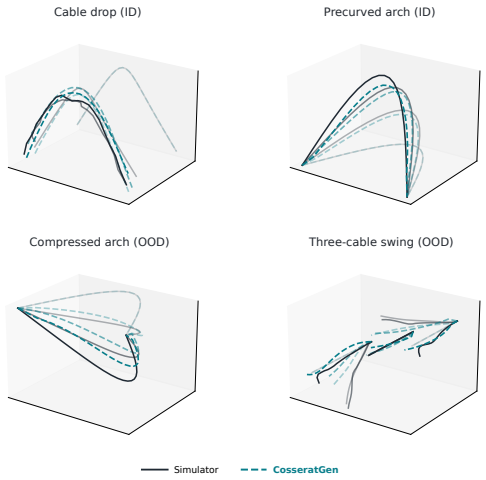

Figure 16: Comparison of DT-NCO predictions and simulator trajectories for ID and composition-OOD dynamics. Teal dashed curves show predicted centerlines and black solid curves show simulator targets at four times. The examples include cable drop and precurved-arch release (ID), and compressed-arch snap and three-cable swing (composition OOD).

Errors are computed in three dimensions. Figure 17 provides the full template-level breakdown.

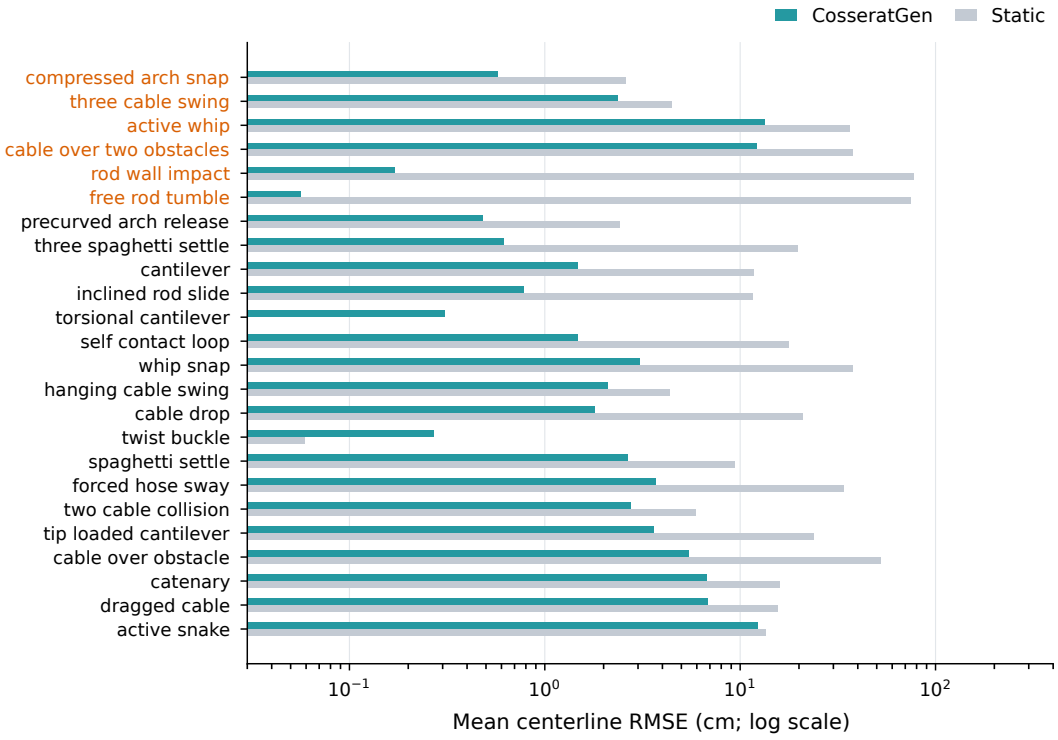

Figure 17: Centerline RMSE (cm, logarithmic axis) by template for 0.8-s rollouts, averaged across seeds 17, 41, and 73. Orange labels mark the six composition-OOD templates; Static repeats the initial state. Lower is better.

Figure 17 reports all 24 templates. Predictions improve over the static reference across all six withheld compositions, with active and contact-rich dynamics remaining the most challenging. These system-level results combine learned prediction with the motion carrier and contact treatment.

H.3 Local Endpoint-Force Control

For the local endpoint-force inverse-control diagnostic, each of ten held-out tip-loaded cantilevers is assigned a target endpoint force of 0.20 or 0.70 N. Starting from 0.45 N, Adam (Kingma & Ba, 2014) optimizes a force constrained to the range from 0.15 to 0.75 N through the frozen operator for 160 steps. We then rebuild and rerun Elastica (Naughton et al., 2021) using only the optimized force, so the curves below show simulator-verified rather than surrogate-only behavior.

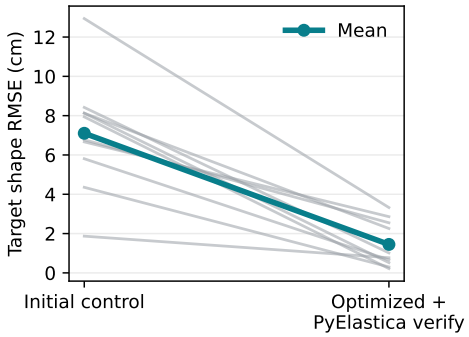

Figure 18: Endpoint-force control of ten held-out cantilevers. Elastica verification before and after optimization (gray: individual cases; teal: mean).

We minimize final-shape RMSE through the operator with other physical conditions fixed. Independent Elastica reruns improve all ten cases (Figure 18), supporting local endpoint-force control.

I Additional Geometries and Clamp Locations

Experimental setup. We examine whether extending training coverage improves prediction for additional initial shapes and clamp locations. This experiment builds on the reconstructed diagnostic in Section G, separately from Cosserat-12K. We consider four conditions: the original planar prebend, a sinusoidal-tangent planar curve, a spatial curve, and the original prebend with the opposite endpoint clamped. Each trajectory contains one strip with 20 elements, released under gravity and recorded at 49 times over 0.8 s. All other physical parameters remain within the original training ranges. For element coordinate $u_j = (j + 1/2)/20$, the new planar tangent is $(\cos\theta_j, \sin\theta_j, 0)$ with $\theta_j = a\sin(2\pi u_j)$; the spatial tangent is proportional to $(1, a\sin(2\pi u_j), a\cos(2\pi u_j))$, with $a \sim \mathcal{U}(0.06, 0.12)$. Both use segments of length $L/20$, parallel-transported material frames, and a straight stress-free reference.

Seventy parameter groups combine four geometries/clamp conditions with four orientations $(0/30/60/90°)$, yielding disjoint sets of 640 training, 160 validation, and 320 test trajectories. We fine-tune for 100 epochs on the new and original training data, without changing architecture or losses. Validation across both datasets and seeds 17/41/73 selects learning rate 3×10^{-4} from $\{10^{-4}, 3 \times 10^{-4}\}$. Both versions use the same held-out test set.

Results. The largest gains occur for the spatial curve and opposite-end clamp, where centerline RMSE decreases from 10.28 to 3.14 cm and from 41.29 to 5.80 cm, respectively. The additional planar curve also improves, from 2.65 to 2.16 cm. These gains come with a tradeoff: error on the original planar geometry rises from 1.75 to 2.05 cm, and spatial-curve stretch error increases despite the lower centerline error. All 1,920 model–trajectory evaluations are finite and included in the aggregates.

J Real-Trajectory Protocol

Table 10 reports the aggregate comparison; the per-object breakdown follows in Table 11.

DEFORM (Chen et al., 2024) contains five real cables and ropes, each recorded at 100 Hz for 350 seconds. We follow its official recording-level split and predict 0.11–0.48 s beyond an observed initial centerline.

Observed endpoint motion is supplied to all variants. Its linear interpolation along the rod is subtracted before residual-deformation prediction and added back for scoring. Real-only and

simulation-pretrained models share real-data windows, normalization, optimization, and model selection.

Table 10: Real-trajectory prediction on DEFORM (70 test clips). RMSE is ordered centerline error in cm, reported as mean±standard error. Const. vel. denotes constant velocity; Final pt. and Arc err. denote final-point and arc errors. The learned variants use DT-NCO. Sim-only uses simulation pretraining without fine-tuning; Real-only trains on real data; Sim+real adds real-data fine-tuning to simulation pretraining. Bold marks the lowest errors.

Method	RMSE↓	Final pt.↓	Arc err.↓
Static	5.07±0.39	7.51	4.74
Const. vel.	8.50±0.66	20.53	25.65
Sim-only	7.92±0.46	15.37	24.10
Real-only	3.34±0.18	5.88	4.84
Sim+real (DT-NCO)	**2.47**±0.12	**4.56**	**4.35**

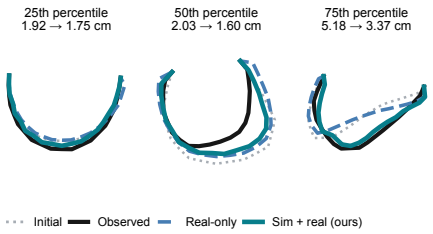

Figure 19: Real-only training and fine-tuning after simulation pretraining on DEFORM under matched real supervision. The three clips are at the 25th, 50th, and 75th percentiles of paired improvement over real-only training. Titles report three-dimensional RMSE; separate PCA views are used only for display.

Table 11: Ordered centerline RMSE (cm) by DEFORM object: 70 official evaluation clips, with 14 per deformable linear object (DLO). The learned variants use DT-NCO. Sim-only uses the frozen simulation-pretrained model; Sim+real adds real-data fine-tuning under the same supervision and optimization as Real-only. Bold marks the lowest error for each object.

Object	Static↓	Sim-only↓	Real-only↓	Sim+real (DT-NCO)↓
DLO1	5.89	8.61	4.05	**2.94**
DLO2	5.32	8.49	3.05	**2.28**
DLO3	4.83	7.47	2.86	**2.16**
DLO4	4.37	7.60	3.18	**2.36**
DLO5	4.93	7.45	3.58	**2.62**

Simulation pretraining followed by real-data fine-tuning lowers error for every object (Table 11). The frozen pretrained model performs worse than Static, indicating the importance of real-data training. Figure 19 illustrates the paired prediction differences.

Real-world prediction to video. Figure 4 uses a separate fine-tuning run (seed 17), not the model in Tables 10 and 11. The held-out DLO4 example was chosen for prediction accuracy and video quality. Initial shape and observed endpoint motion condition prediction; future interior positions are used only for evaluation. One predicted guide drives both appearances with seed 4201 and 30 denoising steps. Its 49 queries span 0–0.48 s; playback at 16 fps slows physical motion by 6.25 times. The unsmoothed marker plot uses displacement from the measured initial position and a shared centimeter scale.

K Video Evaluation Details and Additional Results

We distinguish comparisons across released generators from comparisons of guides within a fixed renderer. The former use each method's supported conditioning format; the latter isolate the effects of evolving structure and interior deformation.

We use Wan2.1-VACE-1.3B (Jiang et al., 2025; Wan et al., 2025) to generate 49 frames at 832×480 and 16 frames per second, with 12 UniPC steps (Zhao et al., 2023) and guidance scale 5. Seeds follow the deterministic schedule $100 + 97i$ for $i \in \{0, \ldots, 71\}$. Static-guide and predicted-guide VACE share camera, guide width, prompt, renderer, and seed; the no-structure baseline uses the

prompt alone. CogVideoX (Yang et al., 2025) uses its trailing diffusion probabilistic model (DPM) scheduler, dynamic classifier-free guidance (CFG), 50 steps, guidance 6, and 49 frames at 720×480 and 8 fps. Outputs are evaluated at 416×240 on a common 49-step normalized-progress axis, rather than equal physical elapsed time.

The trajectory controllers use 50 denoising steps and their native inputs: Tora (Zhang et al., 2025b) receives one free-end track; MagicMotion (Li et al., 2025b) receives a dense predicted tube mask, reference image, and prompt; SG-I2V (Namekata et al., 2025) receives the same reference and five material-point boxes per rod at 14 frames, without text. Controls derive from the same predicted rollout, but their bandwidth differs. Text-conditioned methods share the scene prompt. The comparison evaluates each generator under its native conditioning interface.

Symmetric region-of-interest (ROI) distance averages guide-to-edge and edge-to-guide distances, with the reverse term restricted to a guide-centered corridor. All methods use the same projected simulator target. Table 4 also reports official VBench background consistency and motion smoothness (Huang et al., 2024), and VideoCon-Physics physical entailment from VideoPhy (Bansal et al., 2025).

Additional qualitative protocol. Figures 23–28 compare six scenes spanning gravity-driven deformation, tip loading, and curved release. Each row shows six uniformly sampled frames, with the same scene prompt used by all text-conditioned methods. These additional qualitative runs use a reconstructed full-frame DT-NCO checkpoint and are separate from the aggregate video evaluation above. Ours uses 30 denoising steps; the other generators use 50. Tora receives five tracks sampled along the predicted centerline, while SG-I2V receives one whole-object box whose center follows the mean centerline displacement. MagicMotion receives a dense centerline mask. Both MagicMotion and SG-I2V receive the first frame of Ours as their appearance input; Ours and the text-to-video baselines do not receive a reference image. Columns align normalized video progress rather than physical elapsed time.

K.1 Renderer-Controlled Geometric Baseline

To distinguish a distributed guide from a sparse trajectory control, we construct a privileged endpoint-interpolation baseline on the 12 canonical renderer-matched scenes. It receives both predicted endpoint trajectories for every rod, but no interior state. At normalized initial arc length s, with $0 \le s \le 1$, the interior displacement is the linear blend $(1-s)\Delta\mathbf{r}_{\mathrm{root}}(t) + s\Delta\mathbf{r}_{\mathrm{tip}}(t)$. The projector, guide width, prompt, reference, Wan-VACE weights, 12-step sampler, and diffusion seed are otherwise identical.

Table 12: Distributed versus endpoint-interpolated guidance on 12 renderer-matched scenes. Chamfer, flow endpoint error (EPE), and endpoint distance are in pixels; coverage is the fraction of target pixels within 5 pixels of a rendered edge. Bold marks the best values.

Guide	Chamfer↓	Coverage↑	Flow EPE↓	Flow cosine↑	Endpoint↓
Endpoint interpolation	7.55	0.63	7.44	0.27	**3.64**
Distributed guide	**6.64**	**0.74**	**6.59**	**0.43**	3.77

Table 12 shows better structural and flow agreement with distributed guidance, while endpoint interpolation has slightly lower endpoint error. Figure 20 illustrates the additional interior deformation supplied by the distributed guide.

L Additional Physical-Control Results

The split-level response-fidelity metrics use a common dimensionless motion magnitude. For trajectory i with $K = 49$ frames, valid material nodes $\mathcal{N}_i$, and mean initial rod length L_i, the simulator response is

$$y_i = \frac{1}{L_i}\left(\frac{1}{3K|\mathcal{N}_i|}\sum_{k=0}^{K-1}\sum_{n\in\mathcal{N}_i}\|\mathbf{r}_{i,n}(t_k) - \mathbf{r}_{i,n}(t_0)\|_2^2\right)^{1/2},$$

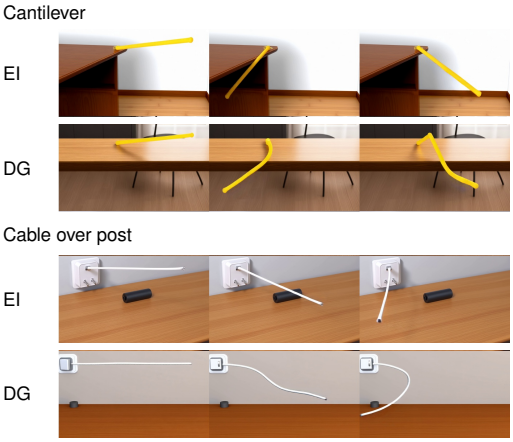

Figure 20: Endpoint-interpolation and distributed guidance for cantilever and cable-over-post scenes. EI: endpoint interpolation; DG: distributed guide. Columns show the initial, middle, and final frames from each video. Both guide constructions use the same predicted endpoint trajectories, with all other generation settings fixed. The distributed guide additionally supplies interior bending and obstacle-mediated shape change.

Endpoint interpolation fixes interior displacement by a linear blend; distributed guidance additionally specifies local bending.

Table 13: Physical-response fidelity and paired control diagnostics for DT-NCO. Pearson r, Spearman ρ, and median absolute relative error (MedARE) compare predicted and simulated motion magnitudes. Endpoint-force optimization is verified by independent Elastica reruns (Naughton et al., 2021).

Split	Pearson $r \uparrow$	Spearman $\rho \uparrow$	MedARE (%)$\downarrow$
ID	0.97	0.98	3.55
Composition OOD	0.60	0.80	14.87
Paired control diagnostics			
Stiffness monotonicity (%)	Prediction / simulator		12.22/50.56
Damping monotonicity (%)	Prediction / simulator		100.00/100.00
Load monotonicity (%)	Prediction / simulator		99.44/97.78
Friction monotonicity (%)	Prediction / simulator		100.00/100.00
Endpoint-force shape RMSE (cm)	Before $\rightarrow$ after		7.10 $\rightarrow$ 1.45
Mean force error (N)	After optimization		0.06

and $\widehat{y}_i$ is computed identically from the predicted centerline. Across the 900 ID and 3,000 composition-OOD trajectories, we report Pearson correlation and Spearman rank correlation (Pearson, 1895; Spearman, 1961), together with

$$\mathrm{MedARE} = 100 \ \mathrm{median}_i \frac{|\widehat{y}_i - y_i|}{\max(|y_i|, 10^{-6})}.$$

The paired parameter sweeps use responses in meters over the 0.8-second rollout. For stiffness and load, the response is the free tip's mean downward displacement from its initial height over $t = 0.4$–0.8 s. For damping, it is the root mean square (RMS) three-dimensional deviation of the free-tip position from its temporal mean over the same interval. For friction, it is the mean material-node displacement between $t = 0$ and $t = 0.8$ s. Adjacent-level monotonicity is the fraction of consecutive level pairs whose predicted response follows the prescribed direction: decreasing for stiffness, damping, and friction, and increasing for load. It is computed within paired configurations before aggregation, so scene variation cannot determine the score.

The sweep covers four parameters at seven levels across 30 paired configurations. Table 13 shows consistent response directions for damping, load, and friction. Stiffness responses are less consistent under the short-window metric, and predicted sensitivity differs from the simulator, including attenuated damping response and overestimated high-friction displacement.

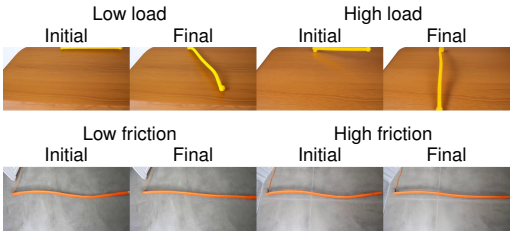

Figure 21: Generated videos under low and high load (top) and friction (bottom), guided by DT-NCO. Each pair shows initial and final frames with the initial state, scene, camera, prompt, and diffusion seed fixed. The operator's load- and friction-dependent response differences remain visible after rendering.

Figure 21 complements the three-dimensional responses in Figure 3 with generated videos.

M Human Evaluation Details

Table 5 in the main paper reports the participant-averaged preferences. Here we detail the protocol, criteria, and aggregation.

Protocol. Following the two-alternative forced-choice (2AFC) protocol in WonderPlay (Li et al., 2025c), we conducted a blinded pairwise study of perceived physics, motion, and appearance. Forty participants each evaluated ten randomized scenes, yielding 400 paired trials across seven baselines and six physical scenario categories in the full study. Table 5 reports seven baseline comparisons. Every trial displayed the scene prompt and two synchronized videos. The reported comparisons pair CosseratGen with CogVideoX-5B (Yang et al., 2025), Wan-T2V-1.3B (Wan et al., 2025), Tora (Zhang et al., 2025b), MagicMotion (Li et al., 2025b), SG-I2V (Namekata et al., 2025), VACE (Jiang et al., 2025) without structure, or static-guide VACE. Method names, guide overlays, and file metadata were hidden. All clips were matched in display size, frame rate, and duration, while trial order and left–right placement were randomized.

Participant instructions. Participants selected one video per criterion in Table 14, without ties.

Table 14: Human-evaluation criteria for each synchronized video pair.

Criterion	Evaluation rule
Physics plausibility	More physically plausible flexible-object behavior under the stated scene condition.
Motion fidelity	More natural and temporally coherent flexible-object motion.
Visual quality	Better appearance and fewer visible rendering artifacts.

The study yields 1,200 criterion-level judgments. For each baseline and criterion, Table 5 averages participant-level preference rates over participants assigned that baseline.

Figure 22 illustrates scene diversity across boundary and load responses, obstacle interactions, orientations, and appearances. Each row shows six uniformly sampled, unretouched frames from one video.

N Physical Consistency Diagnostics

Evaluation setup. We evaluate force balance, moment balance, and energy on 16 trajectories from the reconstructed diagnostic (Section G): four families, two parameter groups per family, and orientations $0°/90°$. All methods share these trajectories and seeds 17/41/73. Metrics are averaged within and then across trajectories; results report the mean and sample standard deviation across seeds.

Force, moment, and energy metrics. We substitute predicted positions and material frames into the native discrete PyElastica balance equations, including external loading, damping, and the rotational transport and dilatation terms. Force and moment errors are the root mean square (RMS) norms of the resulting residuals over sampled times and unconstrained degrees of freedom. The

fixed node and fixed element are excluded because their support reactions are not recorded. All predictors and simulator references use the same five-point temporal differences.

For energy, let $E(t)$ be the sum of translational, rotational, shear/stretch, and bend/twist energies. We measure the work–energy defect

$$b_E(t) = E(t) - E(t_0) - \int_{t_0}^{t} \big(P_{\mathrm{ext}}(s) - P_{\mathrm{damp}}(s)\big)\, ds, \qquad t_0 = 0.02\ \mathrm{s}. \tag{15}$$

Here gravity contributes to external power, not additionally to potential energy; the fixed support performs no work. Under the prescribed damping coefficient ν, $P_{\mathrm{damp}} = 2\nu(K_{\mathrm{trans}} + K_{\mathrm{rot}})$. We report $B_E = \mathrm{RMS}_t(b_E)/(F_0 L)$ and the energy-curve error $C_E = \mathrm{RMS}_t(\widehat{E} - E_{\mathrm{sim}})/(F_0 L)$, where F_0 is rod weight plus the maximum prescribed tip-force magnitude, and L is reference length. These dimensionless metrics are not percentages. B_E assesses energy balance, whereas C_E measures agreement with the simulator's energy evolution.

Validation-selected refinement. Each method receives six 60-epoch refinement trials combining learning rates $\{3 \times 10^{-5}, 10^{-4}\}$ with length/shear/bend loss multipliers $\{1, 4, 16\}$. Architecture and data are unchanged. Seed-17 validation selects the setting, which is repeated for seeds 41 and 73. Eligible trials retain position RMSE within 5% of the starting model; selection minimizes normalized position, shear, and length-scaled bend errors with weights $0.5/0.25/0.25$, including the starting model as a fallback. Training uses AdamW, batch size 64, weight decay 10^{-5}, gradient clipping at 1, and cosine learning-rate decay to one tenth of the initial rate.

Table 15: Physical consistency on 16 reconstructed diagnostic trajectories. Values are mean±sample SD over three seeds. Pos. denotes centerline RMSE (cm); force is in N and moment is in 10^{-2} N m. B_E and C_E are the dimensionless energy-balance and energy-curve errors defined above. Each method uses one validation-selected setting; training budgets are specified in the text. Bold marks the lowest displayed mean in each column.

Method	Pos.↓	Force↓	Moment↓	$B_E \downarrow$	$C_E \downarrow$
FNO	1.15 ± 0.04	18.80 ± 1.76	15.06 ± 0.39	9.27 ± 1.13	9.48 ± 1.13
Transolver++	1.83 ± 0.07	18.25 ± 1.90	16.65 ± 0.54	$\mathbf{8.04} \pm 0.70$	8.21 ± 0.69
DT-NCO	$\mathbf{1.04} \pm 0.06$	$\mathbf{15.64} \pm 2.40$	$\mathbf{12.82} \pm 2.30$	8.10 ± 2.82	$\mathbf{8.07} \pm 2.74$

The DT-NCO row includes an additional six-trial, 120-epoch search over learning rates $\{10^{-5}, 3 \times 10^{-5}\}$ and strain multipliers $\{16, 64, 256\}$, using the same validation rule. It selects 3×10^{-5} and multiplier 16; training and search budgets therefore differ across methods.

Results. Table 15 shows the lowest mean position, force, moment, and energy-curve errors for DT-NCO, while Transolver++ has the lowest energy-balance error. The residuals quantify remaining departures from the dynamics equations.

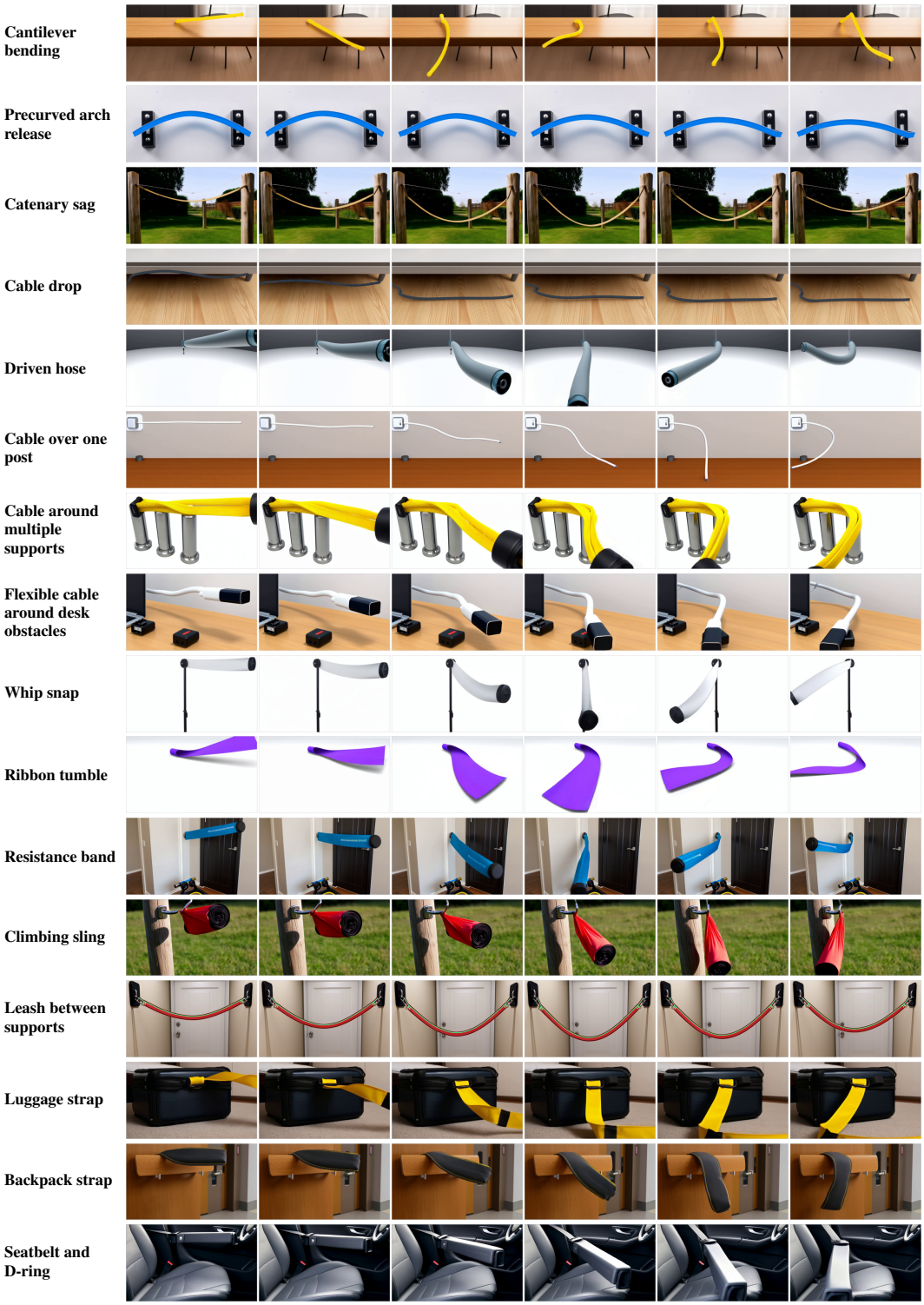

Figure 22: Generated video examples spanning boundary response, external loading, obstacle interaction, orientation changes, and everyday object appearances. Each row shows six uniformly sampled, unretouched frames from one video, ordered by normalized progress from left to right.

O Additional Visual Comparisons

Each row shows six frames in temporal order. Conditioning details are given in Section K.



Figure 23: Visual comparisons on gravity-driven hose bending.

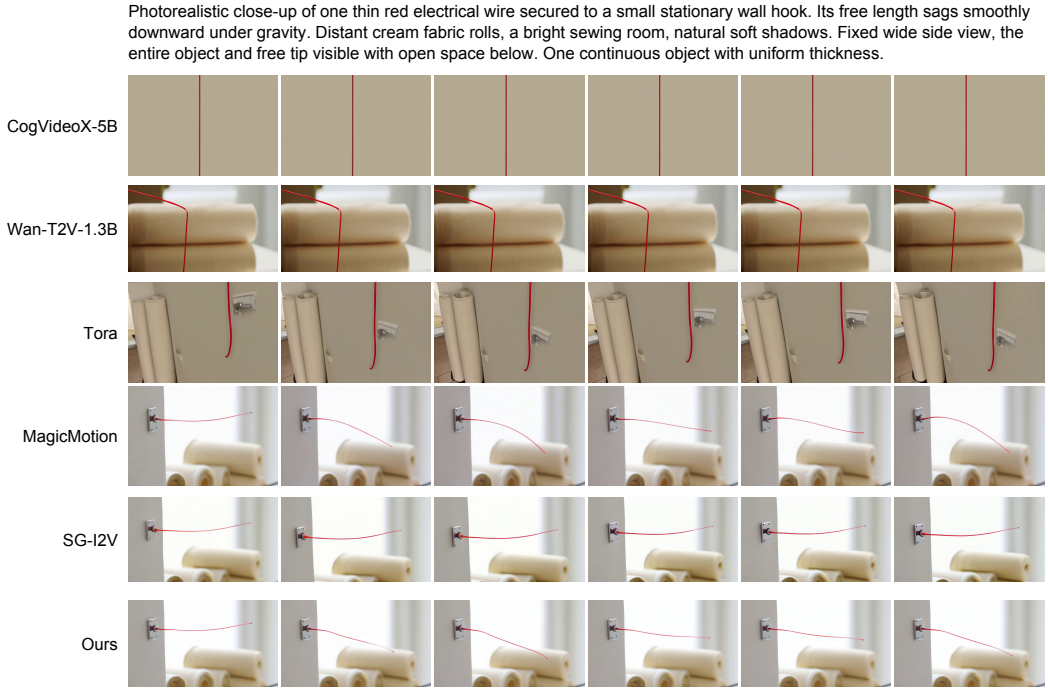

Figure 24: Visual comparisons on gravity-driven wire bending.

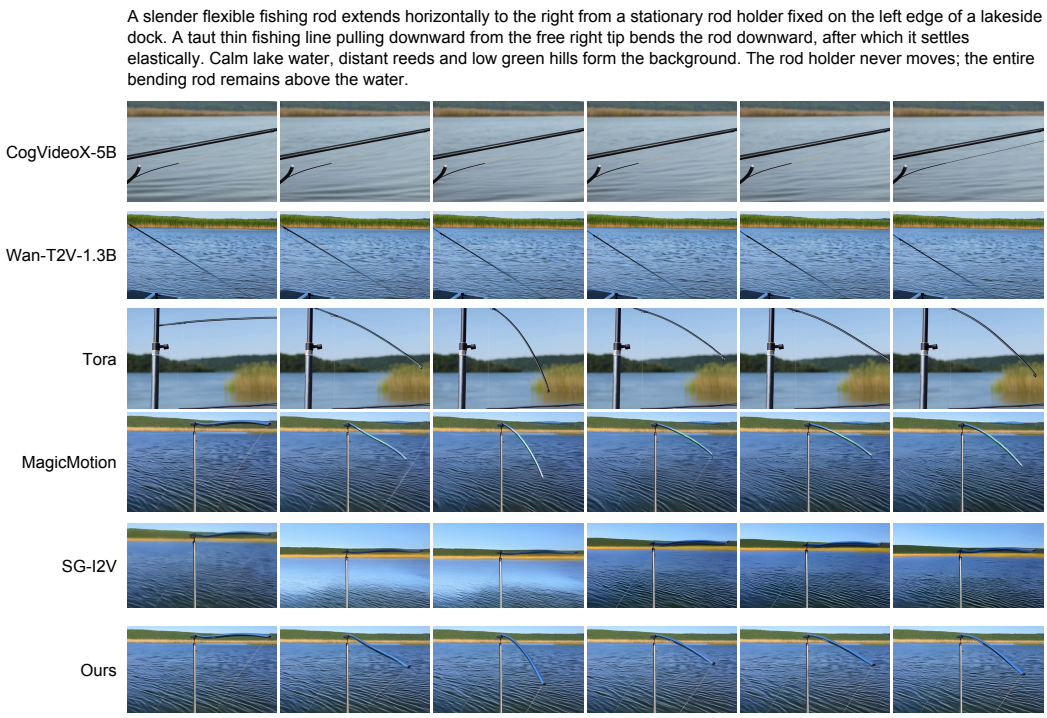

Figure 25: Visual comparisons on tip-loaded rod bending.

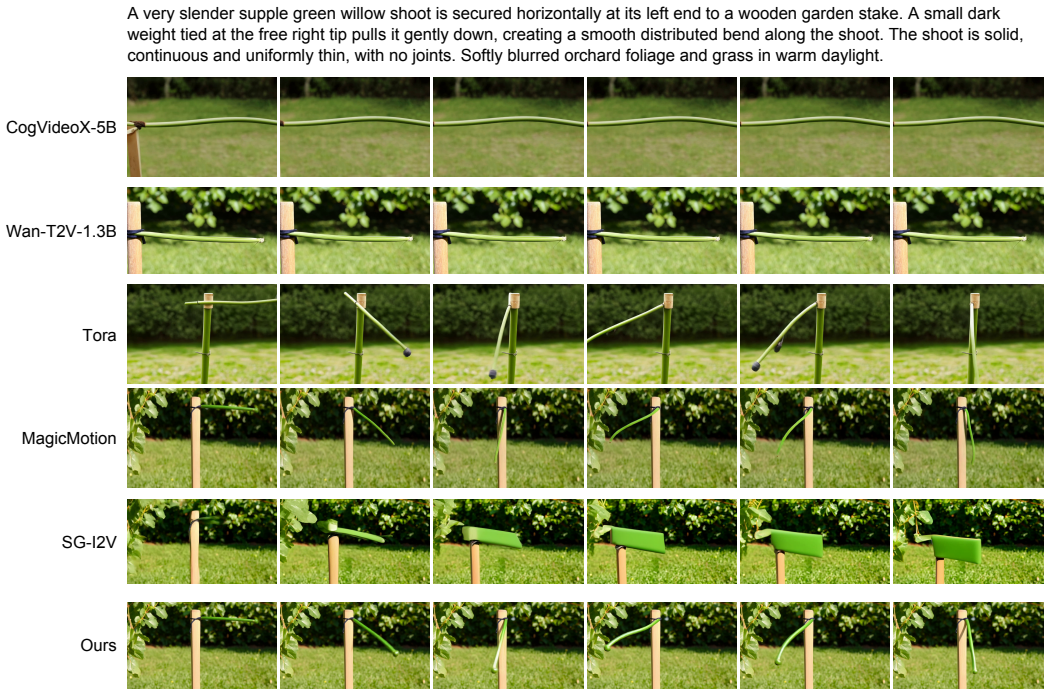

Figure 26: Visual comparisons on tip-loaded branch bending.

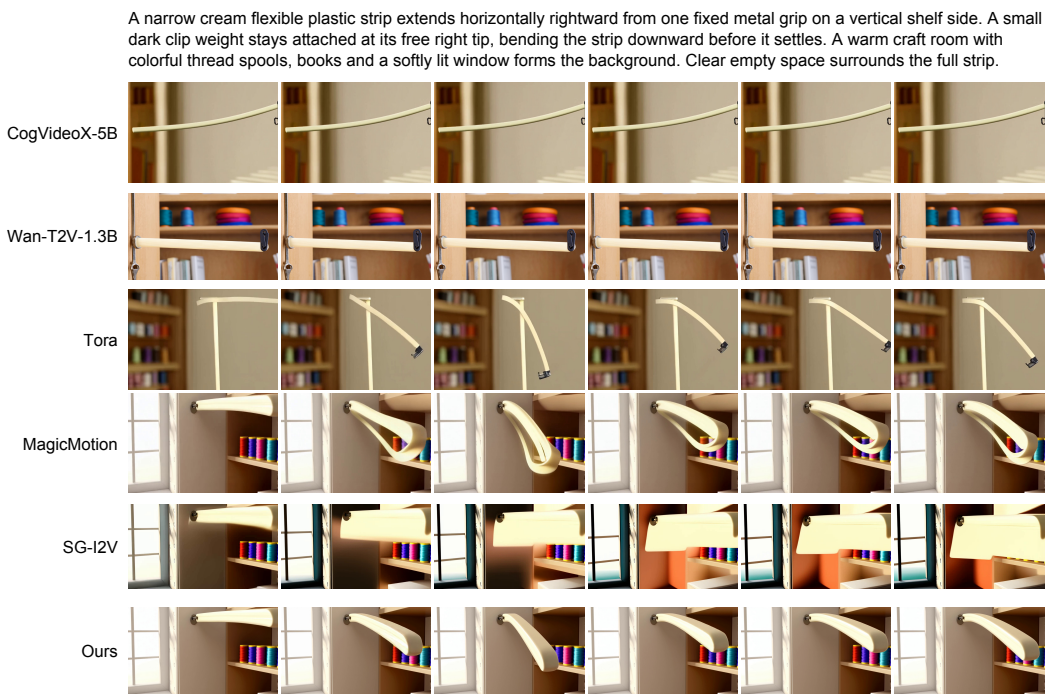

Figure 27: Visual comparisons on tip-loaded flexible strip bending.

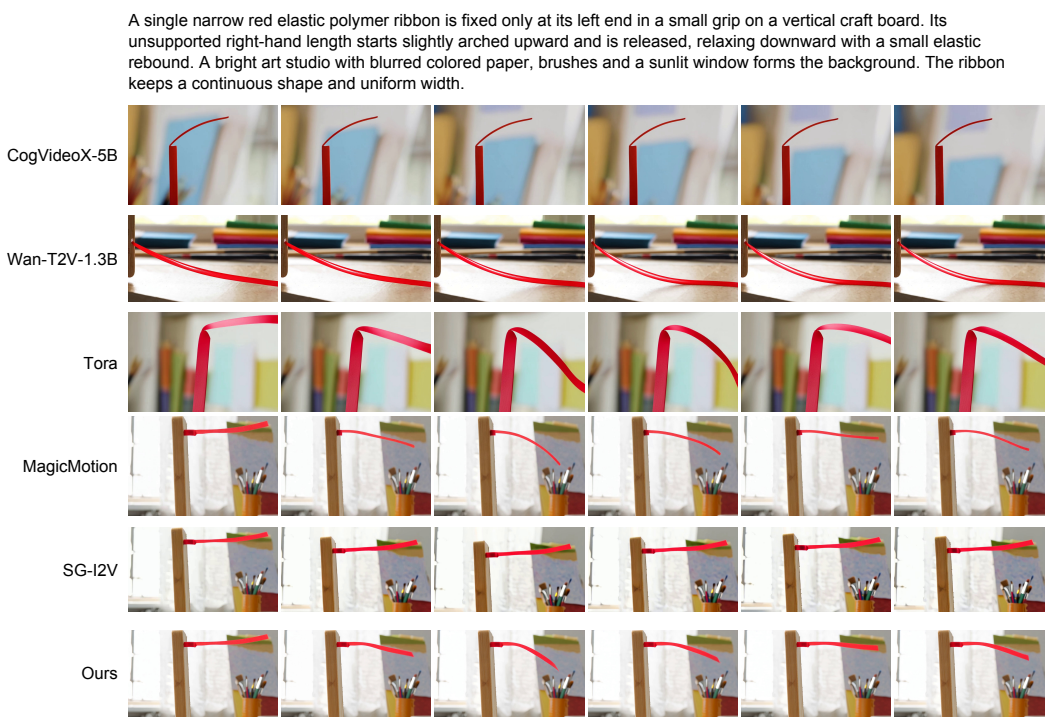

Figure 28: Visual comparisons on curved ribbon release.