Physics-Informed Video Generation via Mixture-of-Experts Latent Alignment

Cong Wang[1,2,3]*, **Hanxin Zhu**[4]*, **Jiayi Luo**[3,5], **Yonglin Tian**[1]
Xiaoqian Cheng[3,4], **Peiyan Tu**[3,6], **Xin Jin**[3,7], **Long Chen**[1]†, **Zhibo Chen**[3,4]†

[1]the State Key Laboratory of Multi-modal Artificial Intelligence Systems,
Institute of Automation, Chinese Academy of Sciences
[2]School of Artificial Intelligence, University of Chinese Academy of Sciences
[3]Zhongguancun Academy
[4]the School of Information Science and Technology, University of Science and Technology of China
[5]Beihang University [6]Zhejiang University [7]Eastern Institute of Technology, Ningbo
{wangcong2024, yonglin.tian, long.chen}@ia.ac.cn
{hanxinzhu, cxq_61, chenzhibo}@mail.ustc.edu.cn
luojy@buaa.edu.cn pytu@zju.edu.cn jinxin@eitech.edu.cn

Abstract

Large-scale video generation models have made remarkable progress in semantic consistency and visual quality, producing videos that are increasingly coherent and visually convincing. Nevertheless, the dynamics induced by pixel-level fitting do not naturally accommodate the regularities that govern real-world motion and interaction, resulting in persistent shortcomings in physical plausibility. To address this limitation, we propose **PILA** (Physics-Informed Latent Alignment), a framework that injects physics-structured latent guidance into the frozen flow-matching dynamics of pretrained video models. Specifically, PILA first employs anchored field estimation to map frozen-generator latents into an operational physical attribute bank organized by field-proxy slots, using observable motion as a kinematic anchor for constructing less directly observed proxies. To handle the heterogeneity of real-world dynamics, PILA adopts a mixture-of-experts design over physical categories. Label-prior masked expert routing selects category-specific operator experts, whose refinements are regularized by operational residuals abstracted from physical relations. Finally, the refined proxies are fused into the physical attribute bank and decoded into a correction to the flow-matching vector field, injecting physics-aware guidance while preserving the visual prior of the pretrained backbone. With staged adapter training on Wan 2.1-1.3B and direct transfer of the learned adapter to Wan 2.2-14B, PILA achieves state-of-the-art results on the physical-plausibility metrics of VBench-2.0, VideoPhy-2, and PhyGenBench, while maintaining competitive visual quality. Project page: https://physwm.github.io/PILA.

1 Introduction

Recent large-scale video generation models have made impressive progress in visual fidelity, enabling the generation of high-quality videos from text prompts [1–4]. Nevertheless, these models still lack reliable mechanisms for representing physical regularities, often yielding collisions with implausible momentum transfer, gravity-inconsistent trajectories, temporally inconsistent material deformations,

*Equal contribution.
†Corresponding author

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

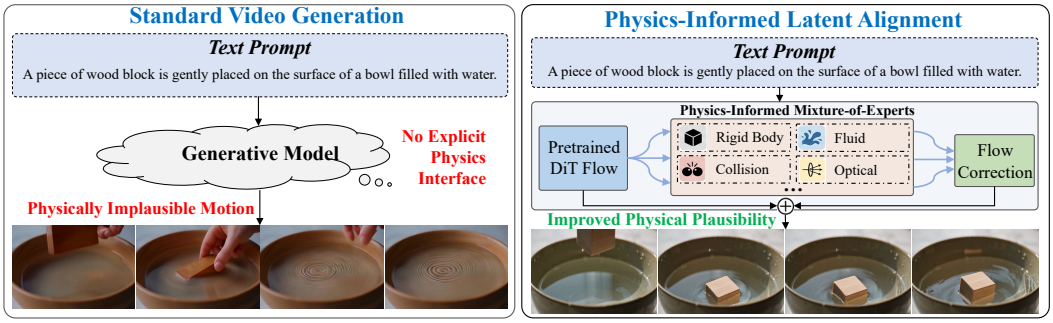

Figure 1: **Standard video generation vs. PILA**. Standard generators may produce physically implausible dynamics. PILA introduces a physics-structured latent adapter with routed experts and flow correction to improve physical plausibility.

and non-causal environmental interactions. Such physical implausibility limits the utility of video generation models in applications that require faithful dynamics, such as embodied-agent training and physical process visualization.

Explicitly integrating physics simulators into generation pipelines [5–7] offers a natural remedy, but also introduces important limitations. These approaches typically require accurate object level 3D geometry [8] and carefully specified physical states, which are costly to obtain and fragile in unconstrained real world settings. Moreover, coupling simulators with high capacity generators adds substantial engineering complexity and remains challenging for realistic foreground and background interactions. This motivates learning physics aware representations directly from video data without explicit simulation. Beyond simulator-based approaches, a growing body of work seeks to improve physical realism through implicit inductive biases in data-driven video generation. Existing methods have explored external physical reasoning and planning to guide synthesis toward more plausible dynamics [9–11], representation-level or inference-time alignment to encourage physical consistency [12, 13], and mixture-of-experts designs to inject structured physical priors into the generation process [14, 15]. Despite their differences, these approaches still introduce physics mainly through prompts, rewards, alignment signals, or coarse expert priors. The unresolved problem is not merely how to add physical supervision, but how to create an operational physical interface, where field-like latent proxies can be constructed, regularized, and propagated back to the latent vector field.

Motivated by this gap, we treat physics-aware generation as proxy-based latent alignment through an operational physical interface inside frozen flow-matching dynamics. Such an interface must construct field-like proxies from generator latents, refine them with category-specific residual structure, and translate the refined physical attribute bank back into the generator's vector field. The objective is not to reconstruct exact simulator states or calibrated physical measurements, but to expose a structured intermediate space where physics-derived constraints can shape a lightweight correction to the pretrained generator.

Physics-Informed Latent Alignment (PILA) instantiates this shared physical interface with an operational physical attribute bank. At each denoising step, a frozen-backbone latent estimate is encoded into fixed field-proxy slots for motion, pressure/density, thermal and phase support, strain/stress, impulse, and wave-like activity. An anchored field estimation (AFE) strategy uses observable motion as a kinematic anchor for completing less directly observed proxies. Label-prior masked expert routing (LPMER) then uses an LLM-assisted label parser to infer prompt-derived physical-category labels and select the corresponding operator experts. Each selected operator applies a recipe-masked residual update to the shared bank, and category-specific residual constraints (CSRC) regularize the corresponding field-proxy representations with operational residuals abstracted from physical relations. These residuals include PDE-style anchors, kinematic consistency terms, latent closure proxies, and stabilizing priors; they are not claimed to be complete governing equations on calibrated physical states. Finally, the refined physical attribute bank is decoded together with the frozen flow-matching prediction into a correction to the flow-matching vector field, injecting physics-aware guidance without replacing the pretrained visual backbone. Our main contributions can be summarized as:

- We propose a multi-expert physical latent-space constraint alignment framework for video generation that can be directly integrated into existing flow-matching generators, injecting physics-aware guidance through a lightweight latent correction.
- We introduce anchored field estimation to construct an operational physical attribute bank from generator latents, and design label-prior masked expert routing to activate the physical-category experts indicated by prompt-derived label priors.
- We propose field-proxy constraint alignment, where each routed expert reads the corresponding field-proxy representation from the attribute bank and is regularized by category-specific operational residuals.
- We demonstrate that the refined physical attribute bank provides an effective physics-aware correction signal for flow matching, yielding state-of-the-art motion continuity and benchmark-measured physical plausibility.

2 Related Works

2.1 Data-driven general purpose video generation

Data-driven general-purpose video generation learns open-domain video synthesis from large-scale video-text and image-video data, and has achieved substantial progress in visual fidelity, semantic alignment, and temporal coherence [1–4, 16]. Modern approaches are typically formulated within latent diffusion [2, 17] or flow-matching [18] frameworks, which support scalable optimization and high-resolution video synthesis, and have increasingly emerged as the backbone of downstream adaptation [19–21] and controllable generation [22–24]. Their strong performance is supported by large and diverse training data [25–28], which expose the models to broad scene distributions, motion patterns, and semantic concepts in real-world videos. However, these models are primarily optimized for appearance quality and semantic consistency, with physical regularities learned only implicitly from data. As a result, they often produce visually convincing videos that still exhibit implausible motion, object interactions, and material dynamics.

2.2 Explicit physics constrained video generation

Explicit physics simulation methods reconstruct 3D scene representations and simulate dynamics using Material Point Methods or Position-Based Dynamics [5, 6, 29–31]. Recent extensions have broadened this paradigm along several directions. WonderPlay [8] and RealWonder [32] combine traditional physics solvers with coarse video generation followed by appearance refinement, aiming to balance physical controllability with visual realism. NeuMA [33] and NewtonGen [34] instead explore neural physics simulation, replacing hand-crafted solvers with learned dynamics models that can be more naturally integrated into generative pipelines. Phys4DGen [35], PSIVG [36], and other methods [37, 38] further extend explicit physical modeling toward 4D simulation and visualization, enriching the representation of dynamic scenes beyond static geometry. Despite these advances, existing explicit simulation based approaches still depend on scene reconstruction, simulator coupling, or specialized physical states, which limits their scalability in open-domain video generation.

2.3 Physics-informed video generation with implicit constraints

Physics-informed video generation can also be pursued without explicit simulation by introducing implicit physical constraints into the generation process. Physical reasoning has been incorporated through chain-of-thought decomposition, reasoning-guided prompt refinement, and staged motion planning to steer synthesis toward physically plausible outcomes [9, 10, 39]. Physical supervision has also been imposed through representation alignment, preference optimization, reward-guided sampling, and world-model-based objectives to encourage physically consistent dynamics during training or inference [12, 40, 13, 41, 42]. Structured physical priors have further been introduced through mixture-of-experts architectures, which specialize different components for distinct physical principles or phenomena [14, 15]. However, these methods still impose physics indirectly through prompts, auxiliary objectives, or coarse expert priors, leaving limited control over how physical constraints are represented and propagated inside the generator. In contrast, our method constructs a shared operational physical interface together with category-specific operators, enabling structured constraint alignment directly in latent dynamics.

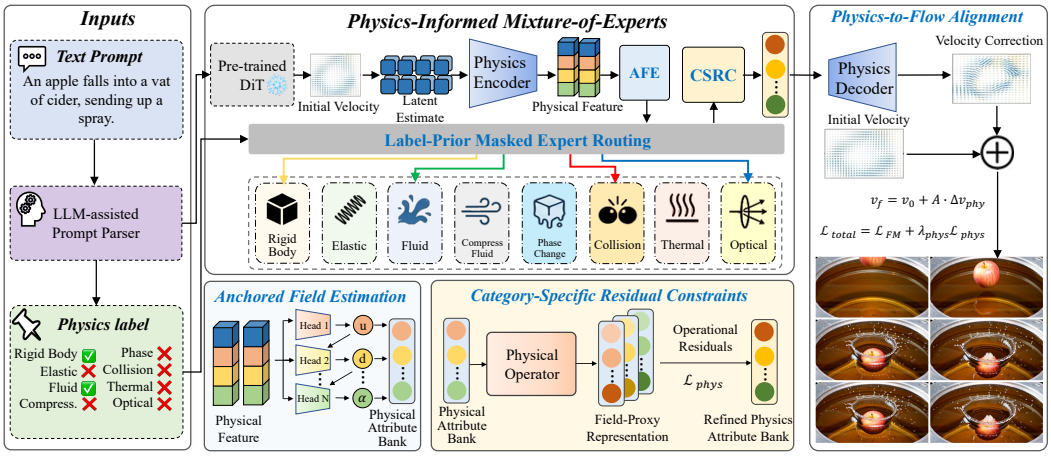

Figure 2: **Pipeline of PILA**. The method proceeds through three stages: (1) operational physical interface encoding and label-prior expert routing (Sec. 3.1); (2) physics-structured operator refinement of category-specific field-proxy representations (Sec. 3.2); and (3) physics-to-latent flow alignment through a lightweight correction to the frozen flow-matching vector field (Sec. 3.3).

3 Methodology

Overview. PILA follows a three-stage information flow. First, a shared physical encoder maps the latent state of a frozen flow-matching generator into a physical feature, which AFE converts into an operational physical attribute bank organized by fixed field-proxy slots, while an LLM-assisted router selects the relevant physical-category experts (Sec. 3.1). Second, the selected operator experts refine this bank through recipe-masked residual updates and category-specific operational residuals (Sec. 3.2). Third, the refined physical attribute bank is decoded into a lightweight latent vector-field correction that aligns the pretrained latent vector field with the routed physical interface (Sec. 3.3).

3.1 Physical Interface Encoding and Expert Routing

Under the flow-matching formulation of the frozen video generator, $v_{\theta_0}(z_t, t)$ denotes the predicted latent velocity for the noisy latent z_t at noise level σ. We then derive a latent-space estimate $\hat{\mathbf{x}}_0$ as the input to the physical encoder:

$$\hat{\mathbf{x}}_0 = z_t - \sigma v_{\theta_0}(z_t, t). \tag{1}$$

The fixed generation condition of the frozen backbone is omitted from v_{θ_0} for notational clarity. This estimate is first encoded into a physical feature:

$$\mathbf{f}_{\text{phys}} = E_{\text{phys}}(\hat{\mathbf{x}}_0, \sigma). \tag{2}$$

The **anchored field estimation (AFE)** module then constructs the operational physical attribute bank. Let $\mathcal{Q}$ denote the set of field-proxy quantities represented by the bank, we obtain the operational physical attribute bank $\mathbf{a}$ by:

$$\mathbf{a} = \text{AFE}(\mathbf{f}_{\text{phys}}) = [\mathbf{a}^{(q)}]_{q\in\mathcal{Q}} \in \mathbb{R}^{32\times T\times H\times W}, \tag{3}$$

where $\mathbf{a}^{(q)}$ denotes the slot of $\mathbf{a}$ associated with quantity q.

The channels of $\mathbf{a}$ are organized into fixed field-proxy slots for displacement, velocity, pressure, density, temperature, phase/support, strain, stress, impulse, and wave-like activity. Since these cues vary in their observability from video, AFE follows an observable-first construction: a motion-derived velocity proxy $\hat{u}$ serves as a kinematic anchor where applicable, while weakly observed variables are introduced progressively through quantity-specific heads. We use "kinematic anchor" for this AFE-internal proxy to distinguish it from the flow-matching latent velocity v_{θ_0}. The resulting slots are operational field proxies used by routed constraints, not exact physical-state estimates. Appendix Sec. A.1 details the field construction used by AFE.

We further introduce a **label-prior masked expert routing (LPMER)** strategy to select category-specific physics operators for the current sample. An LLM-assisted prompt parser first normalizes

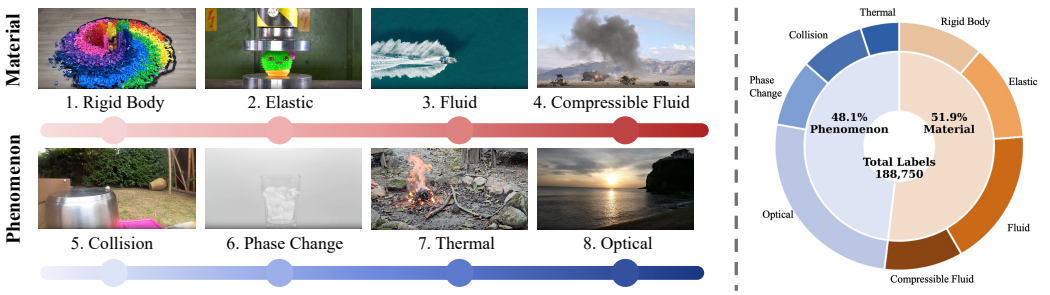

Figure 3: **Training data distribution across physical categories.** We relabel WISA-80K into eight physical categories, grouped into four material categories and four phenomenon categories.

Table 1: **Category-specific operational residual constraints.** For each expert, we list the field proxies and representative residual terms $\mathcal{C}_{k,i}$ used during training.

Physical Category	Proxy Fields	Representative Operational Residuals $\mathcal{C}_{k,i}$
1. Rigid Body	$d, u, \varepsilon, \sigma$	$\partial_t d_k - u_k,\ \varepsilon_k,\ \sigma_k - \gamma_\sigma \varepsilon_k,\ \mathrm{sym}\nabla u_k$
2. Elastic	$d, u, \varepsilon, \sigma$	$\partial_t^2 d_k - c_d^2 \Delta d_k,\ \partial_t^2 u_k - c_u^2 \Delta u_k,\ \varepsilon_k - \mathrm{sym}\nabla d_k,\ \sigma_k - \gamma_\sigma \varepsilon_k$
3. Fluid	u, p, ρ	$\partial_t \rho_k + \nabla\cdot(\rho_k u_k),\ \partial_t u_k + (u_k\cdot\nabla)u_k + \nabla p_k/\rho_k - \nu\Delta u_k,\ \nabla p_k,\ \nabla\rho_k + \mathrm{ReLU}(\rho_0 - \rho_k)$
4. Compress.	u, p, ρ	$\partial_t \rho_k + \nabla\cdot(\rho_k u_k),\ \partial_t(\nabla\cdot u_k) - c_p \Delta p_k,\ p_k - \gamma_p \rho_k,\ \nabla u_k$
5. Phase Change	u, ρ, T, α	$\partial_t \alpha_k + \nabla\cdot(\alpha_k u_k),\ \partial_t T_k + u_k\cdot\nabla T_k - \kappa\Delta T_k - \beta\partial_t\alpha_k,\ \rho_k - \gamma_{\rho\alpha}\alpha_k,\ \nabla\alpha_k$
6. Collision	d, u, j	$\partial_t d_k - u_k,\ \partial_t u_k - \gamma_j j_k,\ \nabla\cdot d_k + \bar{j}_k,\ \Delta j_k$
7. Thermal	u, T	$\partial_t T_k + \nabla\cdot(u_k \bar{T}_k) - \kappa\Delta T_k,\ \nabla T_k$
8. Optical	ψ, α	$\partial_t^2 \psi_k - c_\psi^2 \Delta\psi_k,\ (1-\alpha_k)\psi_k,\ \Delta\alpha_k$

the input prompt into a concise physics-oriented description and extracts a compact set of relevant physical-category labels. The refined prompt description is encoded into a condition vector $\mathbf{c}$, while the extracted category labels are converted into label-prior logits $\mathbf{b}_{\text{label}}$. The router combines learned condition-dependent logits with these label priors to obtain expert routing scores. To balance generation quality and computational efficiency, the model ranks experts by their routing scores and activates a predefined top-k subset:

$$\boldsymbol{\ell} = R_\psi(\mathbf{c}) + \mathbf{b}_{\text{label}}, \qquad \mathcal{K} = \mathrm{TopK}(\boldsymbol{\ell}), \qquad w_k = \mathrm{softmax}(\boldsymbol{\ell}_{\mathcal{K}}/\tau)_k. \tag{4}$$

The selected set $\mathcal{K}$ determines which operator experts may refine the shared physical interface, while the routing weights $\{w_k\}$ determine how their updates are combined downstream. Appendix Sec. A.2 provides additional details on the LPMER routing construction.

3.2 Physics-Structured Residual Modeling

Different physical categories in our taxonomy are associated with different forms of residual structure, such as PDE-style anchors, kinematic consistency terms, latent closure proxies, and stabilizing priors. We therefore introduce **category-specific residual constraints (CSRC)**, which regularize the operational physical attribute bank through residual families associated with the routed categories. Scale factors such as c_d, c_ψ, and γ_σ are latent-space normalizers, not measured material parameters. Each selected expert is implemented as an operator over the shared physical interface: it receives the condition vector $\mathbf{c}$ and the current bank $\mathbf{a}$, then predicts a residual refinement in the same attribute space. A recipe mask $\mathbf{M}_k$ restricts the refinement to the field slots associated with category k:

$$\Delta\mathbf{a}_k = \mathbf{M}_k \odot O_k([\mathbf{c}, \mathbf{a}]), \qquad \boldsymbol{\phi}_k = P_k(\mathbf{a} + \Delta\mathbf{a}_k). \tag{5}$$

Here, O_k is the learnable operator expert for category k, and P_k is a category-specific field readout that extracts the proxy variables required by the corresponding residual constraints. The resulting $\boldsymbol{\phi}_k$ denotes a category-specific field-proxy representation, obtained by reading out the category-relevant slots from the expert-refined physical attribute bank.

The residual objective for expert k is then formulated as:

$$\mathcal{R}_k(\boldsymbol{\phi}_k) = \sum_i s_{k,i} \left\| \mathcal{C}_{k,i}(\boldsymbol{\phi}_k) \right\|^2, \tag{6}$$

Table 2: **Quantitative comparison on VideoPhy [43] dataset.** Joint and Rule are evaluated using VideoPhy-2 [44]. Dynamic, Mechanics, Material, Motion are evaluated using VBench-2.0 [45]. Imaging Quality (Imaging) is evaluated using VBench [46]. The † row transfers the Wan 2.1-trained adapter to Wan 2.2-14B without 14B-specific training. **Bold**: Best. Underline: Second Best.

Method	Joint↑	Rule↑	Dynamic↑	Mechanics↑	Material↑	Motion↑	Imaging↑
Wan 2.1-1.3B [16]	0.643	0.852	0.729	0.518	0.755	0.436	0.589
CogVideoX-5B [4]	0.716	0.902	0.776	0.520	**0.887**	0.459	0.554
VideoREPA-2B [41]	0.684	0.882	0.773	0.521	0.790	0.474	0.519
Ours-1.3B	**0.740**	**0.965**	**0.933**	**0.542**	0.873	**0.576**	**0.600**
Hunyuan-80G [3]	0.546	0.829	0.616	0.551	0.793	0.337	0.637
Wan 2.2-14B [16]	0.690	0.926	0.823	0.563	0.842	0.552	0.653
WISA-14B [14]	0.681	0.897	0.794	0.519	0.820	0.517	0.626
Ours-14B†	**0.743**	**0.953**	**0.953**	**0.608**	**0.907**	**0.651**	**0.658**

where $\mathcal{C}_{k,i}$ denotes the i-th differentiable residual term for category k, and $s_{k,i}$ is its adaptive weight. Table 1 summarizes the field proxies and representative residual families associated with each operator expert. Appendix Sec. A.3 further categorizes these terms as PDE-style anchors, kinematic consistency terms, latent closure proxies, or stabilizing priors, and states the reference relation or prior behind each residual family.

For a sample b, let $\mathcal{K}_b$ be the top-k expert set selected by the router, $\mathcal{Y}_b$ be the set of active physical-category labels predicted from the input prompt, and $\boldsymbol{\phi}_{b,k}$ be the corresponding category-specific field-proxy representation. The routed residual objective is computed only over experts that are both selected and label-consistent:

$$\mathcal{L}_{\text{phys}} = \mathbb{E}_b \left[\frac{1}{Z_b} \sum_{k \in \mathcal{K}_b \cap \mathcal{Y}_b} w_{b,k}\, \mathcal{R}_k(\boldsymbol{\phi}_{b,k}) \right], \tag{7}$$

where $Z_b = \sum_{k \in \mathcal{K}_b \cap \mathcal{Y}_b} w_{b,k}$ normalizes the active routing weights. After applying operational residual constraints to the category-specific field-proxy representations, their residual refinements are aggregated into a refined physical attribute bank:

$$\tilde{\mathbf{a}}_b = \mathbf{a}_b + \sum_{k \in \mathcal{K}_b} w_{b,k}\, \Delta \mathbf{a}_{b,k}. \tag{8}$$

This refined physical attribute bank preserves the shared physical interface while incorporating the category-specific corrections selected by LPMER. Appendix Sec. A.3 provides a term-level interpretation of the residual constraints.

3.3 Physics-to-Latent Flow Alignment

Let $v_{\theta_0}(z_t, t)$ denote a pretrained flow-matching video generator, parameterized as a latent-space vector field. Our goal is to align the frozen generative dynamics with the routed physical interface while preserving the visual prior of the pretrained backbone.

Attribute-to-flow correction. At each denoising step, the frozen backbone first predicts a base vector field $v_{\theta_0}(z_t, t)$. The physics pathway constructs the latent estimate $\hat{\mathbf{x}}_0$, encodes it into the physical feature $\mathbf{f}_{\text{phys}}$, converts the feature into the operational physical attribute bank $\mathbf{a}$ through AFE, and applies routed operator refinements to obtain the refined physical attribute bank $\tilde{\mathbf{a}}$. To transfer this physics-aware representation back to the generative dynamics, we introduce a lightweight attribute-to-flow decoder D_ψ.

The decoder takes the refined physical attribute bank $\tilde{\mathbf{a}}$ as the primary physical input, the task condition vector $\mathbf{c}$ as semantic guidance, and the frozen backbone prediction $v_{\theta_0}(z_t, t)$ as the reference flow direction. The condition vector is broadcast to the spatiotemporal latent resolution and concatenated with the refined physical attribute bank and the base vector field. A small spatiotemporal projection network then maps this refined physical attribute bank to a residual vector field with the same shape as the backbone prediction:

$$\Delta v_\psi = D_\psi([\mathbf{c}, \tilde{\mathbf{a}}, v_{\theta_0}(z_t, t)])\,. \tag{9}$$

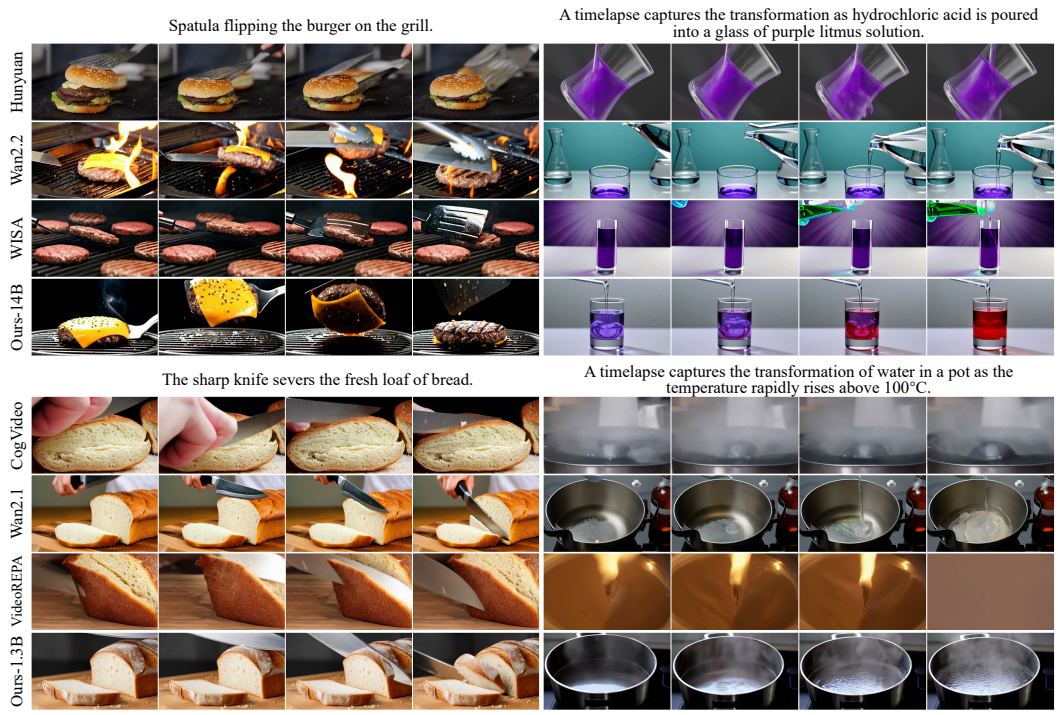

Figure 4: **Qualitative comparison with baselines.** PILA better captures physically grounded interactions and state changes across diverse phenomena.

Here, D_ψ does not decode physical attributes into pixels or explicit physical fields; instead, it converts the refined attribute representation into a latent-space correction that is compatible with the flow-matching vector field.

The physics-aligned vector field is then defined by residual composition:

$$v_{\theta_0,\psi}(z_t, t) = v_{\theta_0}(z_t, t) + \Delta v_\psi. \tag{10}$$

The pretrained parameters θ_0 are kept frozen, and only the physics adapter parameters ψ are optimized. Thus, physics-structured residual constraints affect generation through an additive latent correction, rather than by modifying the pretrained appearance model itself.

Training objective. The full training objective combines flow-matching supervision and routed operational residual constraints:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{FM}} + \lambda_{\text{phys}} \mathcal{L}_{\text{phys}}, \tag{11}$$

where $\mathcal{L}_{\text{phys}}$ is the routed operational residual loss defined in Sec. 3.2.

4 Experiments

4.1 Experimental Setup

Evaluation Dataset. We evaluate on two physics-focused benchmarks, VideoPhy [43] and PhyGenBench [47]. We sample 344 prompts from VideoPhy [43], covering solid-solid, solid-fluid, and fluid-fluid interactions, to evaluate whether generated videos remain physically plausible. PhyGenBench complements this setting with 160 prompts covering 27 physical laws across mechanics, optics, thermal, and material domains, together with domain-wise physics evaluation.

Metrics. To evaluate physical plausibility, we adopt metrics from VideoPhy-2 [44], VBench-2.0 [45], and PhyGenBench [47]. VideoPhy-2 [44] reports a Joint Score (Joint), defined as the proportion of samples whose Semantic Adherence and Physical Commonsense are both greater than or equal to 4. It also evaluates Physical Rules (Rule) by determining whether a video violates predefined physical

Table 3: **Quantitative comparison with baselines evaluated on PhyGenBench [47] dataset.** Mechanics, Optics, Thermal, Material and Average are evaluated using PhyGenBench [47]. Motion is evaluated using VBench-2.0 [45]. Imaging Quality (Imaging) is evaluated using VBench [46].

Method	Mechanics↑	Optics↑	Thermal↑	Material↑	Average↑	Motion↑	Imaging↑
Wan 2.1-1.3B [16]	<u>0.483</u>	0.640	0.322	0.342	0.467	0.250	<u>0.538</u>
CogVideoX-5B [4]	0.483	<u>0.673</u>	0.433	<u>0.425</u>	<u>0.519</u>	0.263	0.508
VideoREPA-2B [41]	0.433	0.627	<u>0.433</u>	0.417	0.490	<u>0.269</u>	0.447
Ours-1.3B	**0.575**	**0.713**	**0.578**	**0.558**	**0.615**	**0.356**	**0.543**
Hunyuan-80G [3]	0.375	0.420	0.211	0.325	0.346	0.150	**0.574**
Wan 2.2-14B [16]	0.458	<u>0.753</u>	<u>0.444</u>	<u>0.417</u>	<u>0.538</u>	0.288	0.559
WISA-14B [14]	<u>0.458</u>	0.700	0.356	0.375	0.494	<u>0.306</u>	0.535
Ours-14B†	**0.625**	**0.840**	**0.589**	**0.617**	**0.683**	**0.381**	<u>0.568</u>

Table 4: **Ablation study on Wan 2.1-1.3B.** All variants use the same backbone and training budget.

Configuration	Mechanics↑	Optics↑	Thermal↑	Material↑	Average↑	Imaging↑	Motion↑
Baseline	0.483	0.640	0.322	0.342	0.467	0.538	0.250
+ Plain Adapter	0.471	0.653	0.502	0.521	0.546	0.534	0.315
w/o Flow correction	0.455	0.563	0.541	0.463	0.507	0.539	0.300
w/o AFE	0.536	<u>0.693</u>	0.486	0.395	0.540	0.537	0.325
w/o MoE operator	0.550	0.637	<u>0.572</u>	0.536	0.578	0.538	0.334
w/o Residual Constraints	0.548	0.685	0.531	0.523	0.582	<u>0.542</u>	0.344
w/o Prompt Refine	0.562	0.668	0.555	<u>0.549</u>	0.591	0.540	<u>0.352</u>
w/o Expert Routing	<u>0.570</u>	0.681	0.571	0.538	<u>0.597</u>	0.541	0.347
Ours-1.3B (Full)	**0.575**	**0.713**	**0.578**	**0.558**	**0.615**	**0.543**	**0.356**

constraints. For VBench-2.0 [45], we adopt Dynamic Spatial Relationship (Dynamic) to assess entity controllability, Mechanics and Material to measure physical plausibility, and Motion Rationality (Motion) to evaluate commonsense reasoning. For PhyGenBench [47], we utilize all available metrics, including Mechanics, Optics, Thermal, and Material. In addition to physics evaluation, we include the Imaging Quality (Imaging) metric from VBench [46] to assess visual quality.

4.2 Implementation Details

We train PILA on top of Wan 2.1-T2V-1.3B [16]. The pretrained video generator is frozen, and only the PILA adapter is optimized on WISA-80K [14] videos with standardized physical-category labels aligned to our four material and four phenomenon categories. Unless otherwise specified, training uses 81-frame videos at 480×832 resolution, a 32-channel operational physical attribute bank, top-4 routed category experts, finite-difference latent residuals, and `bf16` multi-GPU optimization. For Wan 2.2-T2V-A14B evaluation, we do not train a separate adapter; instead, we load the Wan 2.1-trained PILA adapter and apply it as a frozen correction module to the frozen Wan 2.2 backbone. Full training and transfer details are provided in Appendix Sec. B.2.

4.3 Comparison with State-of-the-Art

We compare PILA with recent strong video-generation baselines, including Hunyuan Video [3], CogVideoX [4], Wan [16], VideoREPA [41], and WISA [14]. We report 1.3B–5B and 14B-scale results separately, with Ours-14B† denoting direct transfer of the Wan 2.1-trained adapter to Wan 2.2-14B without additional adapter optimization. Tables 2 and 3 show consistent gains in physical plausibility. On VideoPhy-2, Ours-1.3B improves Joint and Rule by 15.1% and 13.3% over its Wan backbone, respectively. Under the 14B transfer setting, PILA improves all physics-oriented VideoPhy-2 [44] and VBench-2.0 [45] metrics over Wan 2.2-14B, with the largest gains on Motion and Dynamic. On PhyGenBench [47], PILA improves the average score by 18.5% over the strongest 1.3B–5B baseline and by 27.0% over the strongest 14B baseline. Qualitative comparisons in Figure 4

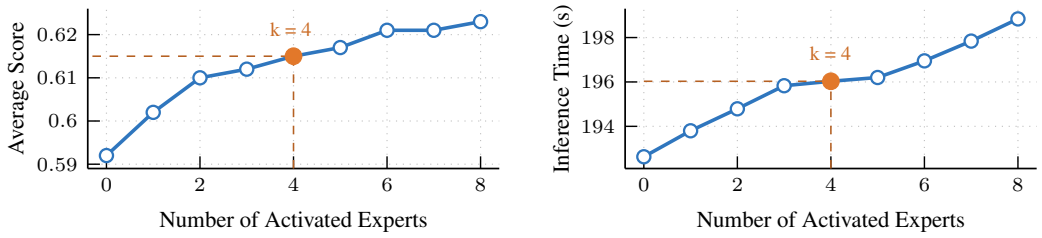

Figure 5: **Quality-efficiency trade-off with different numbers of routed experts.** Orange markers denote the default $k = 4$ setting.

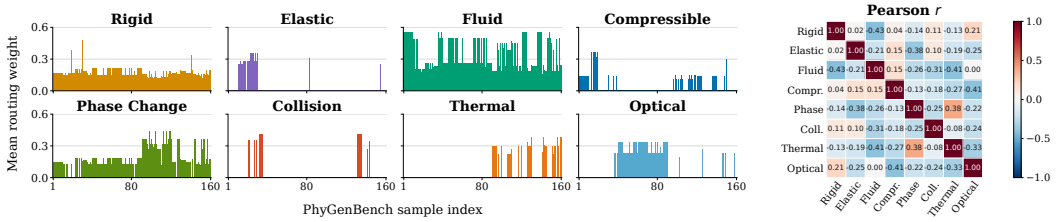

Figure 6: **Analysis of the expert router**. Routing weights are averaged over 50 denoising steps; r denotes their pairwise Pearson correlation across prompts.

further show better object–tool interaction, acid–base color transition, cutting-induced shape change, and boiling dynamics than baselines.

4.4 Ablation Studies

Main component ablations. Table 4 shows that PILA is not merely a capacity effect, as the full model improves the average physics score by 12.6% over the capacity-matched plain adapter. The largest degradation comes from removing the physics-to-flow correction, indicating that the physical bank is most effective when it directly modulates the FM vector field. The drop caused by removing the AFE module further shows that stable physical modeling requires a dedicated latent-to-physics feature interface. More detailed explanation and additional ablations are listed in Appendix Sec. C.1.

Quality-Efficiency Trade-off Study. Figure 5 shows that activating more experts improves the average physics-oriented score, but the gain becomes small after the default setting. Moving from $k = 4$ to $k = 8$ adds only 0.008 score while increasing latency by 2.81s, so we use $k = 4$ as a balanced configuration in all main experiments.

4.5 Analysis of Learned Components

To better understand how PILA internally organizes physical information, we analyze the behavior of its routed experts. Figure 7 visualizes the spatial activation of different experts, showing that the routed branch concentrates corrections on physically active regions rather than applying a uniform global modification. Figure 6 summarizes the step-averaged routing weights across 160 PhyGenBench prompts. The allocations differ across physical-category labels and reflect the combined effects of learned routing scores, label priors, and top-k normalization.

5 Conclusion

In this work, we presented PILA, a multi-expert physical latent-space constraint alignment framework for integrating physics-aware guidance into pretrained flow-matching video generators. PILA constructs an operational physical attribute bank from generator latents through anchored field estimation, and uses label-prior masked expert routing to activate category-relevant physical experts with prompt-derived priors while retaining adaptive routing. Each routed expert reads the corre-

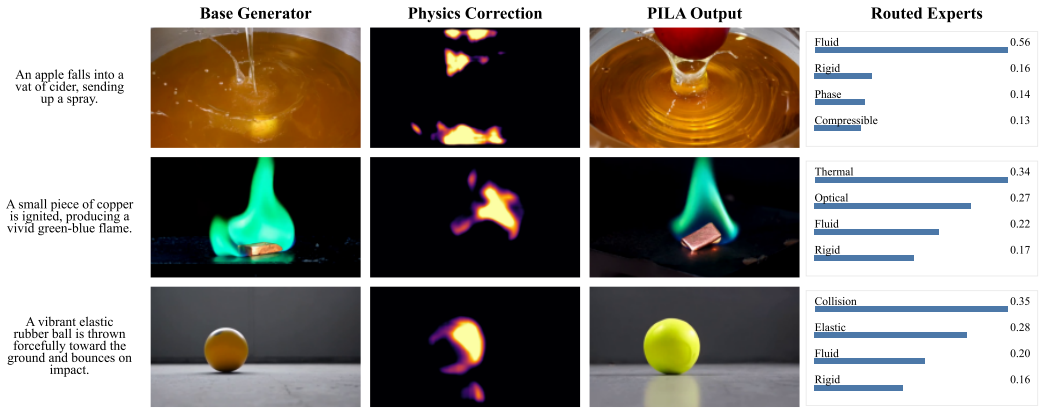

Figure 7: **Visualization of routed expert activations.** Expert maps show that PILA applies category-specific updates to physically active regions.

sponding field-proxy representation from the attribute bank and applies category-specific operational residual constraints, after which the refined bank provides a lightweight correction signal for the flow-matching vector field. Trained on Wan 2.1-1.3B and directly transferred to Wan 2.2-14B, PILA improves motion continuity and benchmark-measured physical plausibility while preserving visual quality. Experiments on VBench-2.0, VideoPhy-2, and PhyGenBench demonstrate state-of-the-art performance, suggesting that operational physical attribute modeling and routed physics-structured alignment provide an effective path toward more physically plausible video generation without requiring calibrated simulator states.

Acknowledgments and Disclosure of Funding

This work was supported by the Zhongguancun Academy Project under Grant No. C20250302 and the Doctoral Student Innovative Research Support Program at the University of Science and Technology of China (Project No. BX-2025-1Y-007). The authors declare no competing interests.

References

[1] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *Advances in neural information processing systems*, 35:8633–8646, 2022.

[2] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023.

[3] Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, et al. Hunyuanvideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024.

[4] 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 *The Thirteenth International Conference on Learning Representations*, 2025.

[5] Tianyi Xie, Zeshun Zong, Yuxing Qiu, Xuan Li, Yutao Feng, Yin Yang, and Chenfanfu Jiang. Physgaussian: Physics-integrated 3d gaussians for generative dynamics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4389–4398, 2024.

[6] Boyuan Chen, Hanxiao Jiang, Shaowei Liu, Saurabh Gupta, Yunzhu Li, Hao Zhao, and Shenlong Wang. Physgen3d: Crafting a miniature interactive world from a single image. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 6178–6189, 2025.

[7] 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*, pages 388–406, 2024.

[8] 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. *arXiv preprint arXiv:2505.18151*, 2025.

[9] Qiyao Xue, Xiangyu Yin, Boyuan Yang, and Wei Gao. Phyt2v: Llm-guided iterative self-refinement for physics-grounded text-to-video generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 18826–18836, 2025.

[10] Yibo Zhao, Hengjia Li, Xiaofei He, and Boxi Wu. Phyrpr: Training-free physics-constrained video generation. *arXiv preprint arXiv:2601.09255*, 2026.

[11] Xindi Yang, Baolu Li, Yiming Zhang, Zhenfei Yin, Lei Bai, Liqian Ma, Zhiyong Wang, Jianfei Cai, Tien-Tsin Wong, Huchuan Lu, et al. Vlipp: Towards physically plausible video generation with vision and language informed physical prior. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12360–12370, 2025.

[12] Jianhao Yuan, Xiaofeng Zhang, Felix Friedrich, Nicolas Beltran-Velez, Melissa Hall, Reyhane Askari-Hemmat, Xiaochuang Han, Nicolas Ballas, Michal Drozdzal, and Adriana Romero-Soriano. Inference-time physics alignment of video generative models with latent world models. *arXiv preprint arXiv:2601.10553*, 2026.

[13] 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.

[14] Jing Wang, Ao Ma, Ke Cao, Jun Zheng, Zhanjie Zhang, Jiasong Feng, Shanyuan Liu, Yuhang Ma, Bo Cheng, Dawei Leng, et al. Wisa: World simulator assistant for physics-aware text-to-video generation. *arXiv preprint arXiv:2503.08153*, 2025.

[15] Zijun Wang, Panwen Hu, Jing Wang, Terry Jingchen Zhang, Yuhao Cheng, Long Chen, Yiqiang Yan, Zutao Jiang, Hanhui Li, and Xiaodan Liang. Prophy: Progressive physical alignment for dynamic world simulation. *arXiv preprint arXiv:2512.05564*, 2025.

[16] 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.

[17] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023.

[18] Yang Jin, Zhicheng Sun, Ningyuan Li, Kun Xu, Hao Jiang, Nan Zhuang, Quzhe Huang, Yang Song, Yadong Mu, and Zhouchen Lin. Pyramidal flow matching for efficient video generative modeling. *arXiv preprint arXiv:2410.05954*, 2024.

[19] Feng Liang, Bichen Wu, Jialiang Wang, Licheng Yu, Kunpeng Li, Yinan Zhao, Ishan Misra, Jia-Bin Huang, Peizhao Zhang, Peter Vajda, et al. Flowvid: Taming imperfect optical flows for consistent video-to-video synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8207–8216, 2024.

[20] Xiaofeng Wang, Zheng Zhu, Guan Huang, Boyuan Wang, Xinze Chen, and Jiwen Lu. Worlddreamer: Towards general world models for video generation via predicting masked tokens. *arXiv preprint arXiv:2401.09985*, 2024.

[21] Tianyu Huang, Wangguandong Zheng, Tengfei Wang, Yuhao Liu, Zhenwei Wang, Junta Wu, Jie Jiang, Hui Li, Rynson Lau, Wangmeng Zuo, et al. Voyager: Long-range and world-consistent video diffusion for explorable 3d scene generation. *ACM Transactions on Graphics (TOG)*, 44(6):1–15, 2025.

[22] Yaosi Hu, Chong Luo, and Zhenzhong Chen. Make it move: controllable image-to-video generation with text descriptions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18219–18228, 2022.

[23] Yuqing Wen, Yucheng Zhao, Yingfei Liu, Fan Jia, Yanhui Wang, Chong Luo, Chi Zhang, Tiancai Wang, Xiaoyan Sun, and Xiangyu Zhang. Panacea: Panoramic and controllable video generation for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6902–6912, 2024.

[24] Xiaoyu Shi, Zhaoyang Huang, Fu-Yun Wang, Weikang Bian, Dasong Li, Yi Zhang, Manyuan Zhang, Ka Chun Cheung, Simon See, Hongwei Qin, et al. Motion-i2v: Consistent and controllable image-to-video generation with explicit motion modeling. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–11, 2024.

[25] Kepan Nan, Rui Xie, Penghao Zhou, Tiehan Fan, Zhenheng Yang, Zhijie Chen, Xiang Li, Jian Yang, and Ying Tai. Openvid-1m: A large-scale high-quality dataset for text-to-video generation. *arXiv preprint arXiv:2407.02371*, 2024.

[26] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3354–3361. IEEE, 2012.

[27] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2446–2454, 2020.

[28] Zhiyu Tan, Xiaomeng Yang, Luozheng Qin, and Hao Li. Vidgen-1m: A large-scale dataset for text-to-video generation. *arXiv preprint arXiv:2408.02629*, 2024.

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

[30] Yuchen Lin, Chenguo Lin, Jianjin Xu, and Yadong Mu. Omniphysgs: 3d constitutive gaussians for general physics-based dynamics generation. *arXiv preprint arXiv:2501.18982*, 2025.

[31] Minh-Quan Le, Yuanzhi Zhu, Vicky Kalogeiton, and Dimitris Samaras. What about gravity in video generation? post-training newton's laws with verifiable rewards. *arXiv preprint arXiv:2512.00425*, 2025.

[32] Wei Liu, Ziyu Chen, Zizhang Li, Yue Wang, Hong-Xing Yu, and Jiajun Wu. Realwonder: Real-time physical action-conditioned video generation. *arXiv preprint arXiv:2603.05449*, 2026.

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

[34] Yu Yuan, Xijun Wang, Tharindu Wickremasinghe, Zeeshan Nadir, Bole Ma, and Stanley H Chan. Newton-gen: Physics-consistent and controllable text-to-video generation via neural newtonian dynamics. *arXiv preprint arXiv:2509.21309*, 2025.

[35] Jiajing Lin, Zhenzhong Wang, Dejun Xu, Shu Jiang, Yunpeng Gong, and Min Jiang. Phys4dgen: Physics-compliant 4d generation with multi-material composition perception. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pages 10398–10407, 2025.

[36] Lin Geng Foo, Mark He Huang, Alexandros Lattas, Stylianos Moschoglou, Thabo Beeler, and Christian Theobalt. Physical simulator in-the-loop video generation. *arXiv preprint arXiv:2603.06408*, 2026.

[37] Hanxin Zhu, Cong Wang, Tianyu He, Long Chen, Xin Jin, Chen Gao, and Zhibo Chen. Cp4d: Compositional physics-aware 4d scene generation. *arXiv preprint arXiv:2606.09187*, 2026.

[38] Shiqian Li, Ruihong Shen, Junfeng Ni, Chang Pan, Chi Zhang, and Yixin Zhu. Learning physics-grounded 4d dynamics with neural gaussian force fields. In *The Fourteenth International Conference on Learning Representations*, 2026.

[39] Zixuan Wang, Yixin Hu, Haolan Wang, Feng Chen, Yan Liu, Wen Li, and Yinjie Lei. Chain of event-centric causal thought for physically plausible video generation. *arXiv preprint arXiv:2603.09094*, 2026.

[40] Cong Wang, Hanxin Zhu, Yonglin Tian, Jiayi Luo, Ruiqi Song, Boyi Sun, Long Chen, and Zhibo Chen. Physflow: Physics-aware optical flow for motion controllable video generation. *arXiv preprint arXiv:2609.08215*, 2026.

[41] Xiangdong Zhang, Jiaqi Liao, Shaofeng Zhang, Fanqing Meng, Xiangpeng Wan, Junchi Yan, and Yu Cheng. Videorepa: Learning physics for video generation through relational alignment with foundation models. *arXiv preprint arXiv:2505.23656*, 2025.

[42] Yuanhao Cai, Kunpeng Li, Menglin Jia, Jialiang Wang, Junzhe Sun, Feng Liang, Weifeng Chen, Felix Juefei-Xu, Chu Wang, Ali Thabet, et al. Phygdpo: Physics-aware groupwise direct preference optimization for physically consistent text-to-video generation. *arXiv preprint arXiv:2512.24551*, 2025.

[43] 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. *arXiv preprint arXiv:2406.03520*, 2024.

[44] Hritik Bansal, Clark Peng, Yonatan Bitton, Roman Goldenberg, Aditya Grover, and Kai-Wei Chang. Videophy-2: A challenging action-centric physical commonsense evaluation in video generation. *arXiv preprint arXiv:2503.06800*, 2025.

[45] Dian Zheng, Ziqi Huang, Hongbo Liu, Kai Zou, Yinan He, Fan Zhang, Lulu Gu, Yuanhan Zhang, Jingwen He, Wei-Shi Zheng, et al. Vbench-2.0: Advancing video generation benchmark suite for intrinsic faithfulness. *arXiv preprint arXiv:2503.21755*, 2025.

[46] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21807–21818, 2024.

[47] 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.

[48] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[49] Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. Zero: Memory optimizations toward training trillion parameter models. In *SC20: international conference for high performance computing, networking, storage and analysis*, pages 1–16. IEEE, 2020.

A Additional Method Details

A.1 AFE Field Construction

This section details how AFE converts the encoder feature into the operational physical attribute bank used by the routed constraints. The construction is intended as an operational latent interface: its slots are field proxies shaped by observable motion and operational residuals, not recovered physical states.

The attribute bank follows a fixed 32-channel contract. Let

$$\mathcal{Q} = (d, u, p, \rho, T, \alpha, \varepsilon, \sigma, j, \psi), \qquad \mathbf{c}_{\mathcal{Q}} = (4, 4, 2, 2, 2, 2, 4, 4, 4, 4), \tag{12}$$

where $\mathbf{c}_{\mathcal{Q}}$ gives the channel width of each slot. The slots correspond to displacement (d), velocity (u), pressure (p), density (ρ), temperature (T), phase (α), strain (ε), stress (σ), impulse (j), and wave (ψ). The encoder first produces $\mathbf{f}_{\text{phys}}$, and a scaffold head maps it to a pre-AFE slot scaffold:

$$\bar{\mathbf{a}} = H_{\text{scaf}}(\mathbf{f}_{\text{phys}}), \qquad \mathbf{a} = \text{AFE}(\bar{\mathbf{a}}, \mathbf{f}_{\text{phys}}) = [\mathbf{a}^{(q)}]_{q \in \mathcal{Q}}. \tag{13}$$

Here, $\bar{\mathbf{a}}$ is an internal implementation scaffold, while the post-AFE bank $\mathbf{a}$ serves as the shared physical interface used throughout the method. The final bank is obtained by reading out a motion-derived velocity proxy as a kinematic anchor, constructing derived kinematic fields, and progressively activating weakly observed slots before repacking the same contract.

Kinematic anchor and descriptors. AFE uses a motion-derived velocity proxy as the primary kinematic anchor where applicable. This terminology separates the AFE-internal proxy $\hat{u}$ from the flow-matching latent velocity v_{θ_0}. The velocity scaffold slot is 4-channel, and a lightweight head reads it into a 2-channel velocity proxy

$$\hat{u} = H_u(\bar{\mathbf{a}}^{(u)}) = (\hat{u}_x, \hat{u}_y). \tag{14}$$

Finite differences on $\hat{u}$ then define the kinematic descriptors used by downstream field constructors:

$$\nabla \cdot \hat{u} = \partial_x \hat{u}_x + \partial_y \hat{u}_y, \qquad \nabla \times \hat{u} = \partial_x \hat{u}_y - \partial_y \hat{u}_x, \tag{15}$$

$$\hat{\varepsilon} = \left(\partial_x \hat{u}_x,\ \frac{1}{2}(\partial_y \hat{u}_x + \partial_x \hat{u}_y),\ \frac{1}{2}(\partial_y \hat{u}_x + \partial_x \hat{u}_y),\ \partial_y \hat{u}_y \right). \tag{16}$$

These descriptors are finite-difference operational fields; they provide local motion geometry rather than exact strain measurements.

Pressure, density, displacement, and stress. Pressure (p) and density (ρ) are constructed from the physical feature and kinematic-anchor descriptors:

$$(\tilde{p}, \tilde{\rho}) = C_{p\rho}([\mathbf{f}_{\text{phys}}, \hat{u}, \|\hat{u}\|, \nabla \cdot \hat{u}, \nabla \times \hat{u}]), \tag{17}$$

$$\hat{p} = \tilde{p} - \text{mean}(\tilde{p}), \qquad \hat{\rho} = \text{softplus}(\tilde{\rho}) + \rho_{\min}. \tag{18}$$

The pressure proxy is centered per sample to remove arbitrary offsets, while the density proxy is mapped to a positive range using a small floor $\rho_{\min}$.

Displacement (d) is constructed by temporal integration of the velocity proxy around the middle frame $t_0 = \lfloor T/2 \rfloor$. With $\hat{d}_{t_0} = 0$, AFE applies midpoint integration forward and backward:

$$\hat{d}_t = \hat{d}_{t-1} + \Delta t_{t-1} \frac{\hat{u}_{t-1} + \hat{u}_t}{2}, \quad t > t_0, \qquad \hat{d}_t = \hat{d}_{t+1} - \Delta t_t \frac{\hat{u}_t + \hat{u}_{t+1}}{2}, \quad t < t_0. \tag{19}$$

The resulting 2-channel displacement proxy is repeated to fill the 4-channel displacement slot.

The stress slot is a closure proxy built from centered pressure and the strain-like descriptor. Let $\text{tr}(\hat{\varepsilon}) = \hat{\varepsilon}_{xx} + \hat{\varepsilon}_{yy}$. AFE uses

$$\begin{aligned} \hat{\sigma}_{xx} &= -\hat{p} + 2\mu \hat{\varepsilon}_{xx} + \lambda \,\text{tr}(\hat{\varepsilon}), \\ \hat{\sigma}_{xy} &= 2\mu \hat{\varepsilon}_{xy}, \qquad \hat{\sigma}_{yx} = 2\mu \hat{\varepsilon}_{yx}, \\ \hat{\sigma}_{yy} &= -\hat{p} + 2\mu \hat{\varepsilon}_{yy} + \lambda \,\text{tr}(\hat{\varepsilon}). \end{aligned} \tag{20}$$

Here, μ and λ are fixed scale factors in the latent proxy space. This relation is used only to provide a stress-like operational field compatible with downstream residuals.

Table 5: **AFE field constructors.** Each constructor produces an operational field proxy that is repacked into the fixed attribute-bank contract.

Slot (meaning)	Constructor	Context	Phase
u (velocity)	$\hat{u} = H_u(\tilde{\mathbf{a}}^{(u)})$	4-channel velocity scaffold slot	`core`
p, ρ (pressure, density)	$(\hat{p}, \hat{\rho})$ from Eqs. 17–18	$\mathbf{f}_{\mathrm{phys}}$, $\hat{u}$, $\|\hat{u}\|$, divergence, curl	`core`
ε (strain)	$\hat{\varepsilon}$ from Eq. 16	spatial finite differences of $\hat{u}$	`core`
σ (stress)	$\hat{\sigma}$ from Eq. 20	$\hat{p}$ and $\hat{\varepsilon}$	`core`
d (displacement)	$\hat{d}$ from Eq. 19	midpoint temporal integration of $\hat{u}$	`core`
α (phase/support)	$\hat{\alpha} = \mathrm{sigmoid}(H_\alpha([\mathbf{f}_{\mathrm{phys}}, \hat{u}, \hat{\rho}, \hat{d}]))$	motion, density, displacement	α
T (temperature)	$\hat{T} = H_T([\mathbf{f}_{\mathrm{phys}}, \hat{u}, \hat{\rho}, \hat{\alpha}])$	motion, density, phase	T
j (impulse)	$\hat{j} = H_j([\mathbf{f}_{\mathrm{phys}}, \hat{d}, \hat{u}, \hat{p}, \hat{\rho}, \hat{\varepsilon}, \hat{\sigma}])$	kinematic and closure proxies	j
ψ (wave)	$\hat{\psi} = H_\psi([\mathbf{f}_{\mathrm{phys}}, \hat{\alpha}])$	physical feature and phase/support proxy; repeated to the 4-channel wave slot	ψ

Progressive recovery of weakly observed slots. Weakly observed variables are introduced progressively. In the only-u recovery policy, the active set grows as

$$\texttt{core} \rightarrow \alpha \rightarrow T \rightarrow j \rightarrow \psi.$$

Table 5 summarizes the constructors. Scalars are repeated to the channel width required by the bank contract, and inactive fields are zeroed before the final contract is packed.

After all active constructors have been evaluated, AFE repacks the constructed field proxies into the fixed slot order in Eq. 12. Fields that are not active in the current recovery phase are explicitly zeroed under the only-u policy. The constructed field proxies are later consumed by CSRC; Sec. A.3 explains how the routed residuals should be interpreted.

A.2 LPMER Routing Details

LPMER connects prompt-level physical semantics to category-specific operators without making routing a fixed single-label decision. We use the physical-category set

$$\mathcal{P} = \{\text{Rigid Body}, \text{Elastic}, \text{Fluid}, \text{Compressible Flow}, \\ \text{Phase Change}, \text{Collision}, \text{Thermal}, \text{Optical}\}$$

where $\mathcal{P}$ denotes the eight active physical-category experts used in the paper. For each sample, the available category evidence is converted into a multi-label set $\mathcal{Y} \subseteq \mathcal{P}$. During training, $\mathcal{Y}$ is read from the relabeled WISA-80K annotations described in Sec. B.1; during inference, the same metadata format can be produced by the LLM-assisted prompt parser after normalizing the prompt into a concise physics-oriented description. This description provides the condition vector $\mathbf{c}$, while the extracted labels define the routing prior. The parser does not supervise the constructed field proxies.

Figures 8 and 9 show the two prompt templates used by this inference-time parser. The first template rewrites the raw generation prompt into a physics-oriented description, and the second template maps that refined description to the active category set $\mathcal{Y}$ used by the LPMER label prior.

The label set is encoded as an additive routing prior rather than as a hard expert mask. Given the condition vector $\mathbf{c}$, the learned router predicts condition-dependent logits

$$\mathbf{r} = R_\psi(\mathbf{c}) \in \mathbb{R}^{|\mathcal{P}|}. \tag{21}$$

The multi-label prior adds a positive bias to the logits of active labels:

$$b_{\mathrm{label},k} = \beta\, \mathbf{1}[k \in \mathcal{Y}], \qquad \boldsymbol{\ell} = \mathbf{r} + \mathbf{b}_{\mathrm{label}}, \tag{22}$$

where β controls the strength of the label prior. This design lets prompt-derived labels guide expert selection while still allowing the learned router to assign capacity to complementary physical categories when their condition-dependent evidence is strong.

LPMER then performs sparse top-k activation:

$$\mathcal{K} = \mathrm{TopK}(\boldsymbol{\ell}), \qquad w_k = \frac{\exp(\ell_k/\tau)}{\sum_{m\in\mathcal{K}} \exp(\ell_m/\tau)}, \quad k \in \mathcal{K}, \tag{23}$$

System Prompt

You rewrite rough prompts into stronger prompts for a text-to-video model.
Preserve the original intent, subject, action, and overall physics.
Make the prompt more concrete by clarifying visible motion, scene layout, camera framing, lighting, and temporal evolution.
Keep the same language as the user's prompt.
Return only a JSON object with one key: refined_prompt.
Example: {"refined_prompt":"..."}

User Prompt

Rewrite this prompt to improve video generation by reconsidering the underlying physical process.
Keep it concise, vivid, and faithful to the original meaning, while strengthening motion consistency and temporal continuity by making the motion more explicit and coherent over time.
Use a single paragraph of 1 to 3 sentences.

Original prompt:
{prompt}

Figure 8: **Prompt-refinement template used before LPMER routing.** The template rewrites a raw input prompt into a concise physics-oriented description that emphasizes objects, motion, interactions, material cues, and state changes. This refined description is used as cleaner prompt evidence for the subsequent category-labeling and label-prior routing steps.

System Prompt

You classify video prompts into physics routing experts.
Return only a JSON object with a single key: labels.
labels must be a non-empty array of 1 to 4 strings chosen exactly from this closed set:
- Rigid Body
- Elastic
- Fluid
- Compressible Flow
- Phase Change
- Collision
- Thermal
- Optical

Rules:
1. Use exact label spellings from the list.
2. Prefer the smallest set of labels that captures the dominant physics.
3. If unsure, return the single best label.
4. Example output: {"labels":["Fluid","Thermal"]}

User Prompt

Prompt:
{refined_prompt}

Figure 9: **LLM-based category-label estimation template for LPMER.** Given the refined physics-oriented description, the template asks the LLM to select the relevant physical categories from the eight-category expert set $\mathcal{P}$. The resulting multi-label set $\mathcal{Y}$ is used as label-prior evidence for routing.

where τ is the routing temperature. In the default setting, $k = 4$, which keeps the physics branch sparse while allowing multi-category scenes to activate more than one operator. This is important for coupled cases such as bouncing, pouring, boiling, or contact-driven deformation, where a single category label would be too restrictive.

Each selected operator receives the shared bank $\mathbf{a}$ and condition vector $\mathbf{c}$, and predicts a residual update restricted by its recipe mask:

$$\begin{aligned} \Delta\mathbf{a}_k &= \mathbf{M}_k \odot O_k([\mathbf{c}, \mathbf{a}]), \qquad \boldsymbol{\phi}_k = P_k(\mathbf{a} + \Delta\mathbf{a}_k), \\ \tilde{\mathbf{a}} &= \mathbf{a} + \sum_{k \in \mathcal{K}} w_k \, \Delta\mathbf{a}_k. \end{aligned} \tag{24}$$

Table 6: **Category-wise glossary of operational constraints.**

Physical Category	Constraint	Name	Role
Rigid Body	$\partial_t d_k - u_k$	Kinematic consistency	Aligns displacement-like and velocity-like slots.
	ε_k	Low-strain prior	Suppresses internal deformation.
	$\sigma_k - \gamma_\sigma \varepsilon_k$	Stress-strain closure	Keeps stress-like and strain-like proxies compatible up to latent scale.
	$\mathrm{sym}\nabla u_k$	Rigid-motion prior	Discourages locally deforming motion while allowing rotation-like components.
Elastic	$\partial_t^2 d_k - c_d^2 \Delta d_k$	Displacement wave	Encourages coherent oscillatory deformation.
	$\partial_t^2 u_k - c_u^2 \Delta u_k$	Velocity wave	Aligns motion with elastic propagation.
	$\varepsilon_k - \mathrm{sym}\nabla d_k$	Deformation compatibility	Couples strain-like and displacement-like proxies.
	$\sigma_k - \gamma_\sigma \varepsilon_k$	Stress-strain closure	Stabilizes elastic state compatibility up to latent scale.
Fluid	$\partial_t \rho_k + \nabla \cdot (\rho_k u_k)$	Mass continuity	Enforces transport-consistent density change.
	$\partial_t u_k + (u_k \cdot \nabla) u_k + \nabla p_k / \rho_k - \nu \Delta u_k$	Momentum balance	Biases velocity toward fluid-style evolution.
	∇p_k	Pressure regularity	Suppresses pressure spikes.
	$\nabla \rho_k + \mathrm{ReLU}(\rho_0 - \rho_k)$	Density stability	Smooths density and discourages collapse.
Compressible Flow	$\partial_t \rho_k + \nabla \cdot (\rho_k u_k)$	Mass continuity	Tracks transported mass variation.
	$\partial_t (\nabla \cdot u_k) - c_p \Delta p_k$	Compression dynamics	Couples local compression changes to pressure-like variation.
	$p_k - \gamma_p \rho_k$	Pressure-density closure	Ties pressure-like and density-like slots together up to latent scale.
	∇u_k	Gradient regularity	Stabilizes expansion and compression patterns.
Phase Change	$\partial_t \alpha_k + \nabla \cdot (\alpha_k u_k)$	Phase transport	Couples support change to local motion.
	$\partial_t T_k + u_k \cdot \nabla T_k - \kappa \Delta T_k - \beta \partial_t \alpha_k$	Thermal-phase coupling	Links heat transport with support change.
	$\rho_k - \gamma_{\rho\alpha} \alpha_k$	Density-phase closure	Keeps density-like and phase/support slots aligned up to latent scale.
	$\nabla \alpha_k$	Support regularity	Stabilizes the phase/support proxy.
Collision	$\partial_t d_k - u_k$	Kinematic consistency	Aligns geometry-like and motion-like slots.
	$\partial_t u_k - \gamma_j j_k$	Impulse response	Couples motion change to impulse up to latent scale.
	$\nabla \cdot d_k + \bar{j}_k$	Contact compression	Encourages compression to match impact.
	Δj_k	Impulse propagation	Smooths impact transfer in space-time.
Thermal	$\partial_t T_k + \nabla \cdot (u_k \bar{T}_k) - \kappa \Delta T_k$	Heat transport	Couples thermal storage, advection, and diffusion.
	∇T_k	Thermal regularity	Discourages isolated temperature spikes.
Optical	$\partial_t^2 \psi_k - c_\psi^2 \Delta \psi_k$	Wave propagation	Encourages coherent optical propagation.
	$(1 - \alpha_k)\psi_k$	Support gating	Keeps wave-like activation inside the support proxy.
	$\Delta \alpha_k$	Support smoothness	Smooths the associated support field.

The mask $\mathbf{M}_k$ is defined by the same field recipe as the residual family of expert k, so routing changes which field-proxy slots can be updated and which residual constraints are eligible downstream. Here, P_k is the category-specific field readout that forms the expert-specific field-proxy representation $\boldsymbol{\phi}_k$ used for residual evaluation. The routed operational residual loss is then evaluated on the label-consistent subset of selected experts, matching the main objective in Eq. 7. Thus, LPMER provides a soft, multi-label routing prior for operator selection, while CSRC supplies category-specific residuals on the corresponding expert-specific field-proxy representations.

A.3 Interpretation of Category-Specific Residual Constraints

We further clarify how to read the operational constraints in Table 1. In the current implementation, these expressions are evaluated on expert-specific field-proxy representations $\boldsymbol{\phi}_k$ rather than on externally supervised physical states. These residuals are not full solver-style governing equations. Instead, CSRC uses four operational constraint families: PDE-style anchors, kinematic consistency terms, closure proxies, and stabilizing priors. PDE-style anchors approximate balance or transport relations, including mass and momentum conservation, heat transport, and wave propagation. Kinematic consistency terms capture definitional links between slots, such as displacement–velocity relations. Closure proxies tie weakly observed slots together up to latent scale, while stabilizing priors suppress high-frequency or unsupported side channels. Table 6 unpacks the main table term by term, while Table 7 states the reference relation or prior from which each operational residual is abstracted.

Table 7: **Residual families and reference physical relations.**

Physical Category	Constraint	Residual family	Reference Equation / Relation
Rigid Body	$\partial_t d_k - u_k$	Kinematic consistency	$\partial_t d = u$
	ε_k	Stabilizing prior	$\varepsilon \simeq 0$ for rigid motion
	$\sigma_k - \gamma_\sigma \varepsilon_k$	Closure proxy	$\sigma \simeq C : \varepsilon$ with latent-scale stiffness
	$\mathrm{sym}\nabla u_k$	Stabilizing prior	$\mathrm{sym}\,\nabla u \simeq 0$
Elastic	$\partial_t^2 d_k - c_d^2 \Delta d_k$	PDE-style anchor	$\partial_t^2 d - c_d^2 \Delta d = 0$
	$\partial_t^2 u_k - c_u^2 \Delta u_k$	PDE-style anchor	$\partial_t^2 u - c_u^2 \Delta u = 0$
	$\varepsilon_k - \mathrm{sym}\nabla d_k$	Kinematic consistency	$\varepsilon = \mathrm{sym}\,\nabla d$
	$\sigma_k - \gamma_\sigma \varepsilon_k$	Closure proxy	$\sigma \simeq C : \varepsilon$ with latent-scale stiffness
Fluid	$\partial_t \rho_k + \nabla \cdot (\rho_k u_k)$	PDE-style anchor	$\partial_t \rho + \nabla \cdot (\rho u) = 0$
	$\partial_t u_k + (u_k \cdot \nabla) u_k + \nabla p_k / \rho_k - \nu \Delta u_k$	PDE-style anchor	$\partial_t u + (u \cdot \nabla) u + \nabla p / \rho - \nu \Delta u = 0$
	∇p_k	Stabilizing prior	bounded pressure-gradient proxy
	$\nabla \rho_k + \mathrm{ReLU}(\rho_0 - \rho_k)$	Stabilizing prior	$\rho > 0$ and locally smooth density
Compressible Flow	$\partial_t \rho_k + \nabla \cdot (\rho_k u_k)$	PDE-style anchor	$\partial_t \rho + \nabla \cdot (\rho u) = 0$
	$\partial_t (\nabla \cdot u_k) - c_p \Delta p_k$	PDE-style anchor	$\partial_t (\nabla \cdot u) - c_p \Delta p \simeq 0$
	$p_k - \gamma_p \rho_k$	Closure proxy	$p \simeq c^2 \rho$ up to latent scaling
	∇u_k	Stabilizing prior	bounded compression gradients
Phase Change	$\partial_t \alpha_k + \nabla \cdot (\alpha_k u_k)$	PDE-style anchor	$\partial_t \alpha + \nabla \cdot (\alpha u) = S_\alpha$ with unresolved source
	$\partial_t T_k + u_k \cdot \nabla T_k - \kappa \Delta T_k - \beta \partial_t \alpha_k$	PDE-style anchor	heat transport with latent phase-change coupling
	$\rho_k - \gamma_{\rho\alpha} \alpha_k$	Closure proxy	$\rho \simeq \rho(\alpha)$ up to latent scaling
	$\nabla \alpha_k$	Stabilizing prior	smooth phase/support proxy
Collision	$\partial_t d_k - u_k$	Kinematic consistency	$\partial_t d = u$
	$\partial_t u_k - \gamma_j j_k$	Closure proxy	$m\, \partial_t u \simeq j$ after absorbing scale
	$\nabla \cdot d_k + \bar{j}_k$	Closure proxy	$\nabla \cdot d + \bar{j} \simeq 0$
	Δj_k	Stabilizing prior	spatial impulse smoothness
Thermal	$\partial_t T_k + \nabla \cdot (u_k \bar{T}_k) - \kappa \Delta T_k$	PDE-style anchor	$\partial_t T + \nabla \cdot (uT) - \kappa \Delta T = Q$ with unresolved source
	∇T_k	Stabilizing prior	bounded temperature-gradient proxy
Optical	$\partial_t^2 \psi_k - c_\psi^2 \Delta \psi_k$	PDE-style anchor	$\partial_t^2 \psi - c^2 \Delta \psi = 0$
	$(1 - \alpha_k) \psi_k$	Stabilizing prior	wave proxy should remain inside active support
	$\Delta \alpha_k$	Stabilizing prior	smooth optical support

1. Rigid Body. The rigid-body expert is derived from rigid kinematics rather than from a deformable material model. The residual $\partial_t d_k - u_k$ comes directly from the displacement–velocity relation $\partial_t d = u$. A rigid body should have negligible internal strain, so ε_k acts as a low-strain prior and $\sigma_k - \gamma_\sigma \varepsilon_k$ keeps the stress-like proxy compatible with that regime up to latent scale. The term $\mathrm{sym}\nabla u_k$ is a finite-difference stabilizing prior for local non-rigidity; using the symmetric part avoids penalizing rotation-like components as strongly as a full velocity-gradient penalty.

2. Elastic. The elastic expert abstracts the small-deformation wave behavior implied by linear elasticity. In a homogeneous medium, the displacement field satisfies a wave-like equation after simplifying the elastodynamic balance; this motivates $\partial_t^2 d_k - c_d^2 \Delta d_k$, and the same propagation bias is applied to the velocity slot through $\partial_t^2 u_k - c_u^2 \Delta u_k$. The term $\varepsilon_k - \mathrm{sym}\nabla d_k$ is an operational compatibility proxy for $\varepsilon = \mathrm{sym}\,\nabla d$, adapted to slot-valued latent fields. The stress-strain term $\sigma_k - \gamma_\sigma \varepsilon_k$ implements a Hooke-style closure $\sigma \simeq C : \varepsilon$ without estimating a full stiffness tensor.

3. Fluid. The fluid expert follows the standard mass and momentum balances used in viscous flow. The continuity residual $\partial_t \rho_k + \nabla \cdot (\rho_k u_k)$ is the conservative form of mass conservation. The momentum residual is a Navier–Stokes-style balance containing temporal acceleration, convective transport, pressure forcing, and viscous diffusion. Since pressure and density are constructed proxies, ∇p_k suppresses pressure spikes and $\nabla \rho_k + \mathrm{ReLU}(\rho_0 - \rho_k)$ encourages smooth positive density. These stabilizers are not additional governing laws; they keep the latent fields numerically usable for residual evaluation.

4. Compressible Flow. The compressible-flow expert keeps mass conservation but replaces the incompressible-style momentum emphasis with terms that expose expansion and compression. The continuity term again comes from $\partial_t \rho + \nabla \cdot (\rho u) = 0$. The residual $\partial_t (\nabla \cdot u_k) - c_p \Delta p_k$ tracks temporal changes of local compression and couples them to pressure-like variation under an acoustic or barotropic linearization. The proxy $p_k - \gamma_p \rho_k$ is a lightweight equation-of-state surrogate,

corresponding to $p \simeq c^2\rho$ up to latent scaling. The gradient term ∇u_k prevents unstable compression patterns from becoming a high-frequency side channel.

5. Phase Change. The phase-change expert is motivated by transported phase indicators and Stefan-style thermal coupling. The phase slot α_k is treated as an occupancy/support proxy, so $\partial_t \alpha_k + \nabla \cdot (\alpha_k u_k)$ links support change to local motion, standing in for an advective phase-balance equation with unresolved source terms. The combined thermal term $\partial_t T_k + u_k \cdot \nabla T_k - \kappa \Delta T_k - \beta \partial_t \alpha_k$ couples storage, advection, diffusion, and latent phase-change activity without claiming calibrated heat capacity or latent heat. The proxy $\rho_k - \gamma_{\rho\alpha}\alpha_k$ ties density to phase occupancy, reflecting that different phases induce different density/support patterns.

6. Collision. The collision expert encodes impulse-response structure rather than solving complementarity contact conditions. The kinematic term $\partial_t d_k - u_k$ again enforces the relation between displacement and velocity. The impulse term $\partial_t u_k - \gamma_j\, j_k$ is derived from the impulse-momentum relation $m\, \partial_t u \simeq j$ after absorbing mass and scale into the latent proxy. The compression term $\nabla \cdot d_k + \bar{j}_k$ encourages converging displacement patterns to coincide with impact activity, while Δj_k smooths the impulse field so that contact response propagates locally instead of appearing as isolated noise.

7. Thermal. The thermal expert uses a compact heat-transport proxy. The term $\partial_t T_k + \nabla \cdot (u_k\, \bar{T}_k) - \kappa \Delta T_k$ combines temporal storage, conservative advection, and diffusion, while ∇T_k limits isolated temperature spikes. We omit unknown heat sources, material heat capacity, and boundary fluxes, so the resulting residual is a low-frequency thermal prior rather than a complete energy-balance solve.

8. Optical. The optical expert abstracts wave-like propagation into a scalar latent proxy. The residual $\partial_t^2 \psi_k - c_\psi^2 \Delta \psi_k$ follows the scalar wave equation obtained after suppressing constants and polarization details. The support-gating term $(1 - \alpha_k)\psi_k$ discourages wave-like activation outside the associated support proxy, while $\Delta \alpha_k$ smooths that support. These terms encourage coherent propagation and support consistency without modeling full Maxwell dynamics.

Unified interpretation. Across all eight physical categories, these terms act as structured inductive bias rather than as certificates of exact physical correctness. They are evaluated by fixed finite-difference operators on expert-specific field-proxy representations, remain tolerant to pseudo-motion noise and missing boundary conditions, and give the frozen generator an interpretable physical interface on which constraint alignment can operate. In that sense, the residual families serve as a practical bridge between constructed field proxies and pretrained generative dynamics.

B Training Details

B.1 Training Dataset and Physical-Category Relabeling

Our training data is based on WISA-80K [14]. We do not merge additional video datasets into the training split. Instead, we reorganize the WISA-80K clips under eight physical categories used by LPMER and CSRC. These categories consist of four material categories (Rigid Body, Elastic, Fluid, and Compressible Flow) and four phenomenon categories (Collision, Phase Change, Thermal, and Optical). This relabeling makes the training annotations consistent with the routed expert taxonomy in Table 1, while preserving the original video-text distribution of WISA-80K. The resulting category distribution is summarized in Fig. 3.

Relabeling protocol. We assign physical-category labels from the video prompt, accompanying physical description, and observable motion pattern. Each clip receives the dominant physical category, and clips containing coupled mechanisms may retain multiple active labels so that LPMER can route several experts. For example, a bouncing ball can activate both Rigid Body and Collision, while boiling water can activate Phase Change, Fluid, and Thermal. The relabeling is therefore mechanism-oriented rather than surface-category-oriented.

Use in training. The relabeled WISA-80K physical-category annotations provide the label priors used by the router and determine which residual families are eligible during physics-coupled training. They do not provide direct supervision for pressure, density, temperature, phase, impulse, or optical

Table 8: **Eight physical categories used to relabel WISA-80K for training.** The taxonomy contains four material categories and four phenomenon categories.

Group	Physical category	Physical criterion	Typical visual cues
Material	Rigid Body	Coherent translation/rotation with negligible internal deformation.	Falling, rolling, sliding, or spinning rigid objects.
Material	Elastic	Reversible deformation with strain and stress-like response.	Bouncing, stretching, bending, oscillation.
Material	Fluid	Liquid-like flow dominated by advection and pressure-driven motion.	Pouring, splashing, dripping, viscous flow.
Material	Compressible Flow	Density-varying gas, smoke, jet, or expansion/compression motion.	Smoke plumes, air jets, explosions, fire-like expansion.
Phenomenon	Collision	Multi-object interaction dominated by impact or contact impulse.	Bouncing contact, stacking, grasping, frictional sliding.
Phenomenon	Phase Change	State transition coupled to thermal or occupancy change.	Melting, freezing, boiling, vaporization, solidification.
Phenomenon	Thermal	Heat transfer or diffusion without a dominant phase transition.	Heating, cooling, thermal convection, glowing heat spread.
Phenomenon	Optical	Light, reflection, refraction, wave, or support-dependent optical effects.	Reflections, lenses, rainbows, caustics, interference-like patterns.

fields. Those fields remain operational proxies constructed through AFE and constrained by the residual terms described in Sec. A.3.

B.2 Implementation Details

Backbones and data. We train the PILA adapter on Wan 2.1-T2V-1.3B [16]. The pretrained video generator is kept frozen, and the training scripts optimize only the PILA adapter. Training clips are read from WISA-80K [14] with the relabeled eight-category annotations described in Sec. B.1. Unless otherwise specified, training videos are resized to 480×832 and sampled as 81-frame clips. For Wan 2.2-T2V-A14B evaluation, we do not optimize a new adapter; we load the Wan 2.1-trained adapter and use it as a frozen correction module on the frozen Wan 2.2 backbone.

Wan 2.1-1.3B training. For the 1.3B backbone, we use a staged schedule. The observable pretraining stage is run over progressively wider diffusion timestep ranges: $[0.90, 1.00]$, $[0.75, 1.00]$, and $[0.50, 1.00]$. Each phase runs for 3K steps with learning rate 1×10^{-5}. The full physics-coupling stage is then trained over $[0.00, 1.00]$ for 3 epochs with the same learning rate. We set the physics loss weight to 0.30, use 2K physics warm-up steps and 1K conditioned-physics warm-up steps, freeze the physical encoder for the first 1K full-coupling steps, and then scale its learning rate by 0.3. The default adapter hidden width is 128, the operational physical attribute bank has 32 channels, and the router selects the top-4 experts.

Wan 2.2-14B transfer. For the 14B backbone, no Wan 2.2-specific adapter training is performed. We directly load the adapter trained on Wan 2.1-1.3B and keep both the Wan 2.2 generator and the adapter weights frozen during evaluation. The transferred adapter keeps the same 32-channel attribute bank, top-4 expert routing, finite-difference residual interface, sigma conditioning, and quadratic sigma gate with floor 0.05. For the Wan 2.2 dual-expert scheduler, the high-/low-noise boundary is set to 0.417 in scheduler-index space when applying the correction across denoising regimes.

Optimization and residual evaluation. The training objective used for the Wan 2.1-1.3B adapter combines flow-matching supervision and routed operational residuals, following the loss in Eq. 11. We optimize trainable adapter parameters using AdamW [48] with a constant learning-rate schedule, `bf16` mixed precision, and DeepSpeed ZeRO-2 [49] with CPU offload. Physics residuals are evaluated by fixed finite-difference operators on expert-specific field-proxy representations rather than by differentiating through a full external simulator. For Wan 2.2-14B, no optimizer steps are taken; the transferred adapter is evaluated directly as an inference-time correction module. During inference, the adapter predicts a lightweight correction to the frozen backbone flow-matching vector field, so the base generator weights remain unchanged.

Compute resources. The Wan 2.1-1.3B adapter is trained on a single 8-GPU node with NVIDIA A100 GPUs, each with 80GB memory. Full WISA-80K training under the staged schedule takes

Table 9: **AFE field-construction ablations on Wan 2.1-1.3B.** We ablate the kinematic anchor and staged recovery used to construct the operational physical attribute bank.

AFE Variant	Mechanics↑	Optics↑	Thermal↑	Material↑	Average↑	Imaging↑	Motion↑
Baseline	0.483	0.640	0.322	0.342	0.467	0.538	0.250
w/o Kinematic Anchor	0.518	<u>0.647</u>	0.553	0.527	<u>0.567</u>	<u>0.542</u>	0.324
w/o Staged Recovery	<u>0.523</u>	0.623	0.545	<u>0.531</u>	0.560	0.541	<u>0.331</u>
PILA (Full AFE)	**0.575**	**0.713**	**0.578**	**0.558**	**0.615**	**0.543**	**0.356**

approximately 40 hours. Unless otherwise stated, the 1.3B ablations use the same hardware configuration and training budget as the full adapter. The Wan 2.2-14B setting is evaluated by direct adapter transfer and does not require additional 14B-specific training.

B.3 Training Procedure

We adopt a staged optimization strategy that separates observable physical encoding, unobserved field-proxy completion, and full flow-matching coupling. This staged training is used for the Wan 2.1-1.3B adapter, which is then transferred to Wan 2.2-14B without additional optimization. This design is critical for preserving the pretrained generator's visual quality while preventing hidden field-proxy slots from becoming unconstrained side channels.

Stage A (Observable Attribute-Bank Pretraining). We first train the shared physical encoder, operational 32-channel attribute-bank head, and observable flow head using frozen-backbone latent estimates and dense flow/deformation proxy targets. The current Wan2.1 training script runs this observable pretraining across progressively wider noise ranges, so the bank learns stable flow-like slots before it is asked to support routed operational residuals.

Stage B (Encoder Completion and Field Recovery). After observable pretraining, we complete the field-proxy interface without updating the generator. AFE follows the field construction described in Sec. A.1: the motion-derived velocity proxy provides the main kinematic anchor, pressure and density are constructed from kinematic descriptors, deformation-related proxies are derived from motion, and weakly observed slots are activated through the ordered phases $\texttt{core} \rightarrow \alpha \rightarrow T \rightarrow j \rightarrow \psi$. The recovery loss combines local only-u fluid terms, kinematic consistency, phase/thermal transport, impulse fitting, wave-propagation anchors, and a small observable anchor. The physical mask used in this stage is built from active routed fields and blended with the bootstrap motion mask.

Stage C (Physics-to-FM Coupling). Finally, the completed encoder initializes the full PILA adapter. The pretrained FM backbone remains frozen; trainable components include the label/sigma condition encoder, top-k router, masked operator experts, shared attribute decoder, and the final correction decoder. The router selects physical categories with label-prior bias, each selected expert emits a recipe-masked attribute update, CSRC regularizes the corresponding expert-specific field-proxy representations, and the refined physical attribute bank is decoded together with v_{θ_0} into Δv_ψ. The full-stage objective combines FM supervision and routed operational residuals on label-consistent expert-specific field-proxy representations.

Practical considerations. In implementation, we apply operational residual constraints in the latent spatiotemporal resolution used by the Wan VAE rather than at pixel resolution, reducing sensitivity to noisy pseudo-motion estimates and preventing over-constraining appearance-level dynamics. The bootstrap motion mask is constructed as a blend of velocity time derivatives (55% weight), latent-space derivatives (25% weight), and spatial velocity gradients (20% weight), then normalized by a per-sample quantile with a floor of 0.08. During full coupling, this mask is blended over the warmup with the active physical mask derived from routed fields. The MoE router uses top-$k = 4$ selection with softmax normalization and label-prior bias. In the main Wan2.1 configuration, the operational bank has 32 channels, the adapter hidden width is 128, the physics weight target is 0.30, the physics warmup is 2000 steps, the conditioning warmup is 1000 steps, and the motion-mask warmup is 300 steps. The overall design provides a stable optimization path that preserves pretrained quality while injecting interpretable physics-aware guidance.

Table 10: **Operational residual-family ablations on Wan 2.1-1.3B.** Each row removes one residual subfamily, using the grouping clarified in Tables 6–7, while keeping the same backbone and evaluation protocol.

Configuration	Mechanics↑	Optics↑	Thermal↑	Material↑	Average↑	Imaging↑	Motion↑
Baseline	0.483	0.640	0.322	0.342	0.467	0.538	0.250
Proxy-only fields	0.518	0.647	0.543	0.527	0.565	0.541	0.306
w/o PDE-style anchors	0.539	0.679	0.550	0.544	0.586	0.539	0.345
w/o closure proxies	0.552	0.691	0.546	0.543	0.592	0.535	0.353
w/o kinematic consistency	0.547	0.696	0.552	0.548	0.595	0.542	0.349
w/o stabilizing priors	0.563	0.665	0.576	0.542	0.593	0.540	0.354
Full Model	**0.575**	**0.713**	**0.578**	**0.558**	**0.615**	**0.543**	**0.356**

Table 11: **Additional routing-label ablations on Wan 2.1-1.3B.** All variants use the same backbone and evaluation protocol.

Configuration	Mechanics↑	Optics↑	Thermal↑	Material↑	Average↑	Imaging↑	Motion↑
Baseline	0.483	0.640	0.322	0.342	0.467	0.538	0.250
w/ Random Label	0.500	0.687	0.570	0.542	0.582	0.532	0.307
w/ LLM label	**0.575**	**0.713**	**0.578**	**0.558**	**0.615**	**0.543**	**0.356**

C Additional Experiment Details and Analysis

C.1 Supplementary Ablation Analysis

This section expands the main ablation study into module-level controls. All variants use the Wan 2.1-1.3B backbone and the same evaluation protocol. Table 4 reports the main component ablations, while Tables 9–11 give the focused analyses for the three core modules: AFE field construction, CSRC residual supervision, and LPMER label-prior routing. The goal is to isolate where the physical improvements come from: added adapter capacity, the operational physical interface, category-specific operator refinement, residual supervision, routing-label quality, and the final physics-to-flow injection.

Explanation of key component ablations. Table 12 defines each variant used in Table 4. The plain adapter control separates PILA from a capacity-only explanation. The AFE and flow-correction ablations evaluate the contributions of physical attribute-bank construction and attribute-to-flow correction, respectively. The MoE-operator, residual-family, and expert-routing controls then isolate the middle of the pipeline, where category-specific updates and residual losses are applied to the shared bank.

Ablation of anchored field estimation (AFE) strategy. Table 9 expands the AFE ablation beyond the single main-table component removal. It asks whether the operational physical bank should be built from an observable kinematic anchor and whether weakly observed fields should be introduced through staged recovery. Removing the kinematic anchor still improves over the frozen baseline because the adapter can learn generic corrections, but it lowers the average physical score from 0.615 to 0.567 and the motion score from 0.356 to 0.324. Removing staged recovery similarly reduces the average score to 0.560. These drops indicate that AFE is not simply a free 32-channel side branch: the motion-derived proxy grounds the downstream field constructors, and staged recovery prevents pressure, density, thermal, phase, impulse, and wave slots from becoming unconstrained latent channels. Imaging quality remains nearly unchanged across these variants, so the AFE gain mainly comes from better physical organization rather than a visual-quality trade-off.

Ablation of label-prior masked expert routing (LPMER) strategy. Table 11 focuses on the label-prior side of LPMER. Random labels intentionally break the semantic correspondence between prompts and routed experts. This setting still improves over the frozen baseline because the adapter and expert pathway remain active, but it trails the LLM-label setting in both average physical score (0.582 vs. 0.615) and motion score (0.307 vs. 0.356). The gap shows that LPMER is not only adding sparse expert capacity; it benefits from physically meaningful prompt-derived category evidence. This table complements the "w/o Expert Routing" and "w/o Prompt Refine" rows in Table 4: those

Table 12: **Definitions of ablation variants.**

Variant	Changed component	Diagnostic purpose
Baseline	Uses the frozen Wan 2.1-1.3B generator without the PILA adapter.	Measures the pretrained backbone before adding any trainable physics pathway.
+ Plain Adapter	Keeps an extra trainable adapter but removes the structured physical bank, routed operators, and residual constraints.	Tests whether gains come mainly from added capacity.
w/o Flow correction	Disables the attribute-to-flow correction.	Evaluates the contribution of attribute-to-flow correction to physical plausibility.
w/o AFE	Disables the anchored field estimation (AFE) strategy described in Sec. A.1.	Evaluates the contribution of anchored field estimation to constructing the operational physical attribute bank.
w/o MoE operator	Removes category-specific masked operator refinement.	Tests whether routed operator specialization matters beyond a shared physical bank.
w/o Residual Constraints	Removes CSRC residual supervision during training.	Tests whether operational residuals shape the adapter beyond FM supervision alone.
w/o Prompt Refine	Disables prompt refinement while retaining physical-category label estimation for routing.	Evaluates the contribution of prompt refinement to physical consistency.
w/o Expert Routing	Disables sparse top-k expert selection and uses a non-routed expert path.	Tests whether label-prior routing is useful after the physical bank is present.
w/o Kinematic Anchor	Removes the motion-derived velocity proxy used as the primary AFE construction anchor.	Tests whether AFE needs an observable kinematic anchor before constructing derived field proxies.
w/o Staged Recovery	Uses a non-staged field recovery path rather than the observable-first AFE recovery schedule.	Tests whether progressively activating weakly observed slots is needed for stable field construction.
w/ Random Label	Replaces physical-category labels with random labels for routing.	Tests whether the router gains come from meaningful physical labels rather than label noise.
w/ LLM label	Uses LLM-inferred physical-category labels for routing.	Tests whether prompt-level physical semantics provide useful routing priors.
Full Model	Uses AFE, LPMER, masked operators, CSRC, and physics-to-flow correction.	Reports the complete PILA configuration used in the main comparison.

rows test whether sparse routing and prompt refinement are useful, while Table 11 isolates whether the routed labels themselves carry useful physical semantics.

Ablation of category-specific residual constraints (CSRC) strategy. Table 10 decomposes CSRC into the residual families used by the routed experts. The proxy-only variant keeps the constructed field proxies but removes the structured residual-family decomposition; it improves over the no-adapter baseline but remains far below the full model (0.565 vs. 0.615 average), showing that proxy fields alone are insufficient. Removing PDE-style anchors, closure proxies, kinematic consistency terms, or stabilizing priors each reduces the average score relative to the full CSRC objective. The effect is distributed rather than dominated by one term family: PDE-style anchors support transport, diffusion, momentum, heat, and wave behavior; closure proxies couple stress, strain, pressure, density, phase, and impulse proxies up to latent scale; kinematic consistency keeps motion and displacement-like fields coherent; and stabilizing priors suppress noisy finite-difference artifacts and unsupported side channels. The full CSRC combination gives the best score on every physics metric while preserving imaging quality, supporting the claim that the residuals act as complementary constraints on the same operational physical interface.

Denoising-step schedule. Figure 10 visualizes the denoising-step sensitivity reported in Table 13. As the correction range expands from late-only steps to the full 1–50 schedule, physics-oriented average and motion scores improve steadily, while imaging quality remains nearly unchanged. This trend supports the sigma-conditioned and sigma-gated correction strategy: early noisy states are handled conservatively, but allowing the adapter to operate throughout denoising lets physical alignment accumulate across the sampling trajectory.

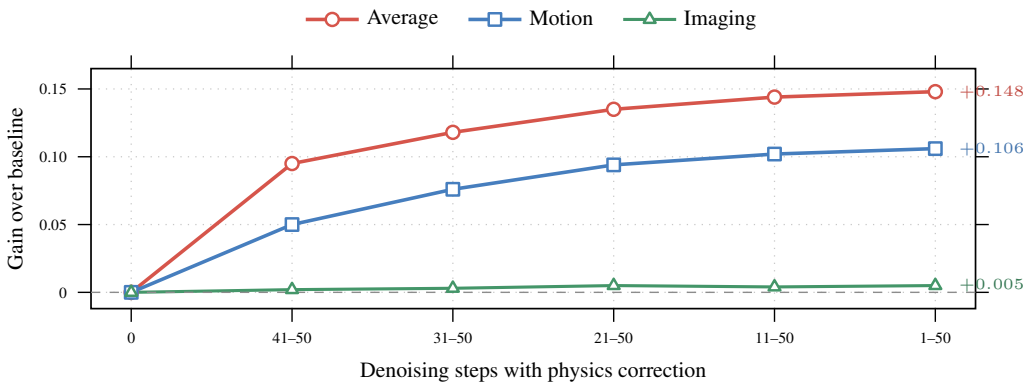

Figure 10: **Effect of the denoising-step schedule.** Gains are computed relative to the no-adapter baseline in Table 13. Expanding the sigma-gated correction window improves physical average and motion scores while preserving imaging quality.

Table 13: **Denoising-step sensitivity of the physics adapter on Wan 2.1-1.3B.** We vary the denoising-step range in which sigma-gated physical corrections are applied while keeping the same backbone and evaluation protocol.

Adding Steps	**Mechanics↑**	**Optics↑**	**Thermal↑**	**Material↑**	**Average↑**	**Imaging↑**	**Motion↑**
0	0.483	0.640	0.322	0.342	0.467	0.538	0.250
41-50	0.535	0.673	0.500	0.492	0.562	0.540	0.300
31-50	0.558	0.695	0.540	0.520	0.585	0.541	0.326
21-50	0.567	0.704	0.560	0.540	0.602	0.543	0.344
11-50	<u>0.572</u>	<u>0.710</u>	<u>0.570</u>	<u>0.552</u>	<u>0.611</u>	<u>0.542</u>	<u>0.352</u>
1-50	**0.575**	**0.713**	**0.578**	**0.558**	**0.615**	**0.543**	**0.356**

Top-k routing sensitivity. Table 14 gives the numeric values behind Fig. 5. Increasing k activates more physical-category experts and gradually improves the average physics-oriented score, but the gain becomes small after $k = 4$. The marginal increase from $k = 4$ to $k = 8$ is modest relative to the additional latency, so the main experiments use $k = 4$ as the default quality-efficiency balance.

Table 14: **Sensitivity to top-k expert selection.**

k	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Average↑	0.592	0.602	0.610	0.612	0.615	0.617	0.621	0.621	0.623
Time (s)	192.63	193.80	194.79	195.83	196.03	196.20	196.95	197.84	198.84

Router and expert-map diagnostics. The router analysis in Fig. 6 and the expert-map visualization in Fig. 7 are diagnostic complements to the ablation tables. The router statistics check whether selected experts follow the relabeled physical-category taxonomy, while the expert maps check whether routed updates are spatially concentrated on physically active regions. These diagnostics are not separate training settings; they explain how the full model uses the mechanisms ablated above.

C.2 Qualitative evidence.

Figures 11 and 12 provide additional qualitative comparisons for the 1.3B–5B and high-capacity settings, respectively. They show that PILA more consistently follows physically plausible object interactions and temporal evolution under the same prompts, complementing the quantitative comparisons in Tables 2 and 3. Figure 13 further shows category-wise PILA generations across the eight physical categories used by LPMER and CSRC. These examples illustrate the breadth of the generated phenomena, while also motivating the limitations discussed below.

C.3 Baseline Details

We compare PILA with the baselines reported in Tables 2 and 3. The comparison is organized by model scale because video-generation quality and physical plausibility are both strongly affected by backbone capacity. All methods are evaluated on the same benchmark prompt sets and with the same metric pipelines described in Sec. 4; no baseline is given access to PILA's operational physical attribute bank or routed residual losses.

Backbone baselines.

- **Wan 2.1-1.3B** [16]: the frozen text-to-video backbone on which the 1.3B PILA adapter is trained. This row measures the base generator before adding the explicit physical interface, LPMER routing, CSRC residuals, and physics-to-flow correction.
- **Wan 2.2-14B** [16]: the large-scale Wan backbone used as the transfer target in our 14B setting. We include it to separate the effect of model scale from the effect of the transferred PILA adapter.

General text-to-video baselines.

- **CogVideoX-5B** [4]: a strong open text-to-video diffusion model with an expert-transformer design. It serves as a mid-scale general-generation baseline in the 1.3B–5B comparison group.
- **Hunyuan Video-80G** [3]: a large-scale video generation system included in the high-capacity comparison group. This baseline helps assess whether PILA's physics gains are explained purely by larger model capacity.

Physics-aware video-generation baselines.

- **VideoREPA-2B** [41]: a physics-oriented video generation method that aligns generated dynamics with relational signals from foundation models. We include it as a mid-scale physics-aware baseline.
- **WISA-14B** [14]: a physics-aware video generation baseline built around world-simulation assistance. It is included in the 14B-scale group and is particularly relevant because our training data is based on WISA-80K, while PILA uses the data through a relabeled eight-category taxonomy rather than through WISA's model architecture.

PILA variants. Ours-1.3B denotes PILA trained on the Wan 2.1-1.3B backbone with the full physical interface and routed residual objective. **Ours-14B**† denotes the transfer setting: the adapter trained on Wan 2.1-1.3B is loaded onto the frozen Wan 2.2-14B backbone without Wan 2.2-specific adapter optimization. The dagger is used throughout the tables to mark this no-retraining transfer protocol.

D Limitations and Discussions

Limitations. PILA improves physical plausibility by aligning a pretrained flow-matching generator with an operational physical attribute bank, but the bank is designed as a latent interface rather than a full simulator state. Its field slots provide physics-aware proxies shaped by observable motion anchors and operational residuals, which is appropriate for video generation but does not aim to recover calibrated material parameters or exact physical measurements. The category-specific residual constraints also use compact approximations of physical relations, leaving detailed boundary conditions, source terms, and 3D geometry to future extensions.

Discussion and future work. Future work can build on PILA in two directions. First, richer cues such as depth, object tracks, 3D correspondence, or simulator-assisted annotations could further calibrate the physical attribute bank. Second, learned or parameterized physical operators could extend the residual families to more scene-specific boundary conditions and material properties.

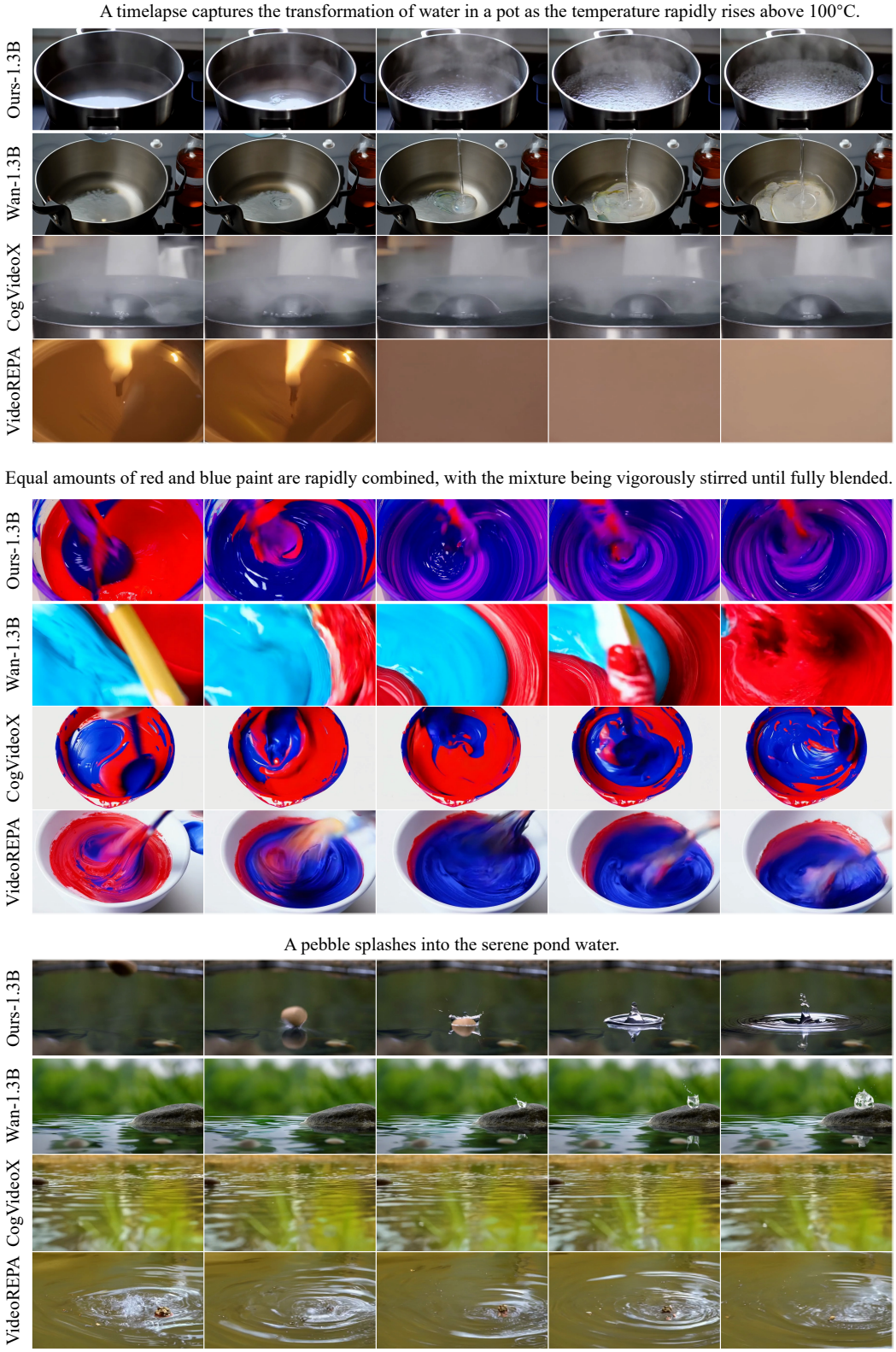

Figure 11: **Qualitative comparison in the 1.3B–5B setting.** PILA produces more physically plausible motion and interactions under the same prompts.



Figure 12: **Qualitative comparison in the high-capacity setting.** PILA produces more plausible physical evolution under the same prompts.



Figure 13: **Additional PILA generations across physical categories.** Each row shows sampled frames from one physical category in our eight-category taxonomy.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: The abstract and introduction state the method components and empirical scope, including PILA's physical interface, AFE, LPMER, CSRC, and physics-to-flow correction. The claims are supported by the experiments in Sec. 4 and scoped by the limitations in Sec. D.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: We discuss limitations in Sec. D.

   Guidelines:

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

3. **Theory assumptions and proofs**

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

Answer: [N/A]

Justification: The paper does not present formal theoretical theorems or proofs. The equations define the model, routing, losses, and operational residual constraints.

Guidelines:

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

4. **Experimental result reproducibility**

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

Answer: [Yes]

Justification: The method, losses, expert taxonomy, evaluation metrics, training data source, training schedule, and transfer protocol are described in Secs. 3–4 and Appendix Secs. B.1–B.3. These details specify the experimental protocol supporting the main claims.

Guidelines:

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

5. **Open access to data and code**

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

   Answer: [Yes]

   Justification: The training data and evaluation benchmarks are based on publicly available cited resources. We will publicly release the code, training and inference scripts, and trained PILA adapter weights to support reproduction of the main experimental results.

   Guidelines:

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

6. **Experimental setting/details**

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

   Answer: [Yes]

   Justification: Sec. 4 gives the evaluation datasets, metrics, baselines, and main implementation setup. Appendix Secs. B.1, B.2, and B.3 provide training data, relabeling, optimizer, schedules, hyperparameters, and transfer details.

   Guidelines:

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

7. **Experiment statistical significance**

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

   Answer: [No]

   Justification: The paper reports benchmark scores and ablations under fixed evaluation protocols, but it does not report error bars, confidence intervals, or repeated-run statistical tests. Running repeated video-generation evaluations for all baselines and ablations would be computationally expensive.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.

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

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: Appendix Sec. B.2 reports the compute setup: the Wan 2.1-1.3B adapter is trained on 8 NVIDIA A100 GPUs with 80GB memory each, and full WISA-80K training takes approximately 40 hours. It also states that 1.3B ablations use the same setup and that Wan 2.2-14B transfer does not require additional training.

   Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: To the best of our knowledge, the research conforms to the NeurIPS Code of Ethics. It uses cited pretrained models, public benchmarks, and WISA-80K-derived training data, and it does not involve human-subject experiments or private user data.

   Guidelines:

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

10. **Broader impacts**

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

Answer: [N/A]

Justification: The work is a methodological study of physics-informed latent alignment for video generation and is not tied to a specific deployment, user-facing application, or release of a new pretrained generator or dataset. We therefore do not identify a distinct societal impact beyond the general context of video generation research.

Guidelines:

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

11. **Safeguards**

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

Answer: [N/A]

Justification: The submission does not release a new pretrained video generator, scraped dataset, or high-risk model asset. PILA is evaluated as an adapter on existing cited backbones.

Guidelines:

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

12. **Licenses for existing assets**

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

Answer: [Yes]

Justification: The paper properly credits the datasets, benchmarks, pretrained models, and evaluation suites used in the experiments. All existing assets are used in accordance with their original licenses and terms of use, and no third-party asset is redistributed or relicensed in a way that violates those terms.

Guidelines:

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

13. **New assets**

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

Answer: [N/A]

Justification: The paper does not release a new standalone dataset, benchmark, or model asset in the current submission. It documents the WISA-80K relabeling protocol and method details in Appendix Secs. B.1 and B.2.

Guidelines:

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

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

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

Answer: [N/A]

Justification: The paper does not involve crowdsourcing experiments or research with human subjects. The experiments use existing video-generation benchmarks and datasets.

Guidelines:

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

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

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

    Answer: [N/A]

    Justification: The paper does not involve human-subject research, crowdsourcing, or participant studies. Therefore IRB or equivalent human-subject review is not applicable.

    Guidelines:

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

16. **Declaration of LLM usage**

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

    Answer: [Yes]

    Justification: The paper describes LLM usage in the LPMER routing pipeline, where an LLM-assisted parser produces prompt-derived physical-category evidence for expert routing. This usage is described in Sec. 3.1 and Appendix Sec. A.2, including the prompt-refinement and category-label estimation templates.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.