This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

NS-Diff: Fluid Navier–Stokes Guided Video Diffusion via Reinforcement Learning

Zijun Deng and Yuxin Peng*
Wangxuan Institute of Computer Technology, Peking University
dengzijun@stu.pku.edu.cn, pengyuxin@pku.edu.cn

Abstract

*While recent video generation models achieve impressive visual quality, generating physically plausible videos remains challenging, especially for fluid dynamics and rigid-body motions. To address this, we present **NS-Diff**, a physics-guided reinforcement learning framework for video diffusion. First, we design a noise-robust physical dynamics detector that distinguishes rigid and fluid regions by analyzing motion in noisy latent frames. Second, we introduce a Physics-Conditioned Latent Injection module, which encodes velocity fields, deformation gradients, and material masks, and injects them into the DiT denoiser via cross-attention. Third, we introduce a reinforcement learning optimization module that enforces simplified Navier-Stokes constraints on fluid dynamics and minimum-jerk principles on rigid bodies through policy gradients. Experiments on PhysVideoBench, UCF, and MSR-VTT show that our approach reduces jerk errors by 43%, decreases fluid divergence by 33%, and improves FVD by 22.7%, achieving higher physical plausibility and visual quality.*

1. Introduction

Recent advancements in video generation, largely driven by the development of diffusion-based architectures [14, 27, 33], have led to models like SORA [4], which achieve remarkable visual fidelity and long-term coherence. However, as demonstrated in Fig. 1, these models frequently generate physically implausible motions despite their visual quality. For instance, rigid objects in generated videos may deform like soft bodies [4]. Without incorporating physical constraints, video generation models remain vulnerable to producing artifacts that break physical realism, highlighting the need for methods that integrate physics-aware principles to improve motion credibility.

Integrating physics into video generative models is challenging. Existing approaches either extract physical cues without explicit modeling, or rely on computationally expensive simulations and manually predefined constraints. For example, VideoJam [6] captures motion and physical representations from video sequences but does not enforce governing physical laws. Some methods, such as Xu et al. [43], enforce physical consistency in 3D shape generation by explicitly modeling mechanical properties, and Yuan et al. [47] use motion imitation in a physics simulator to guide human motion synthesis. These explicit modeling approaches typically require extensive simulation data or manually defined constraints, limiting their generalizability. Inspired by advances in reinforcement learning, it is possible to optimize structured behaviors and enforce constraints without relying on predefined simulations.

To address these challenges, NS-Diff introduces a reinforcement learning framework that integrates lightweight physics constraints directly into the diffusion denoising process. For rigid bodies, motion smoothness is optimized by minimizing Jerk (the rate of change of acceleration) using motion dynamics extracted from noise-robust optical flow. For fluids, simplified Navier-Stokes constraints with divergence and smoothness regularization maintain realistic flow continuity. This RL-based approach not only allows us to effectively model rigid body dynamics and fluid behavior, but also enables our method to mitigate a wide range of general physics violations such as object disappearance, emergence, splitting, or merging. These behaviors are inconsistent with motion mechanics and are discouraged during reinforcement learning.

The framework consists of three key components. 1) **Physical Attribute Detection**: Identifies rigid and fluid regions from noisy latent frames. 2) **Physics-Conditioned Latent Injection**: Encodes velocity fields, deformation gradients, and material masks into latent features and injects them into the DiT denoiser. 3) **Reinforcement Learning Optimization**: Refines motion through policy gradients, enforcing Jerk minimization for rigid bodies and divergence–smoothness regularization for fluids. An adaptive activation mechanism ensures physics guidance is applied only when motion cues are reliable.

* Corresponding author.

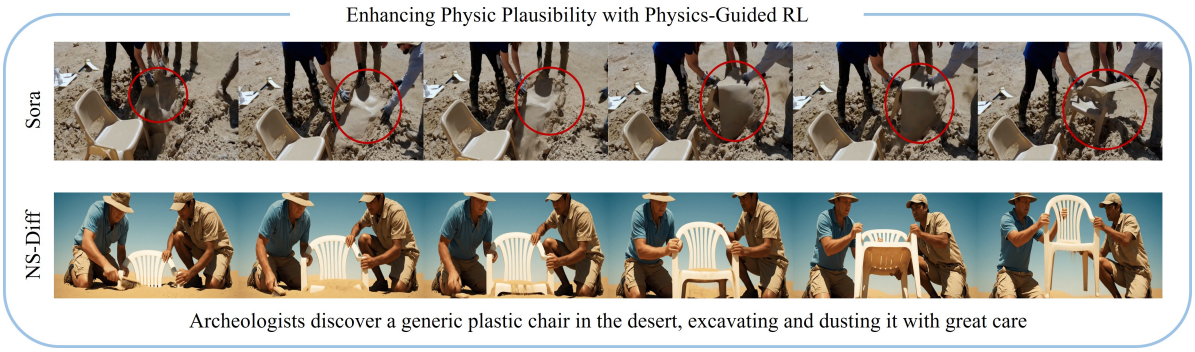


Figure 1. Existing methods like Sora [4] introduce physically implausible deformations (highlighted by red circles), while NS-Diff preserves rigid structures (the plastic chair) and more realistic fluid-like sand behavior.

To evaluate the effectiveness of our approach, we construct PhysVideoBench, a new dataset designed to assess the physical accuracy of video generation models. Existing benchmarks primarily focus on perceptual quality, lacking dedicated metrics for physics-based evaluation. PhysVideoBench fills this gap by providing curated sequences with ground-truth motion data, enabling rigorous assessment of physical plausibility in generated videos. Our contributions are threefold:

- Introduction of **NS-Diff**, a Navier–Stokes guided video diffusion framework optimized via reinforcement learning.
- Development of a Physics-Conditioned Latent Injection and noise-robust attribute detection, enabling accurate rigid-body and fluid motion modeling from latent representations.
- Construction of PhysVideoBench, a benchmark for evaluating physical consistency in generated videos.

Experiments on PhysVideoBench, UCF-101, and WebVid show that our method not only improves rigid and fluid motion fidelity but also mitigates general unphysical artifacts and enhances temporal motion consistency, demonstrating the effectiveness of physics-guided reinforcement learning in video synthesis.

2. Related Work

2.1. Physics-aware generative modeling

Physics-aware generative modeling aims to improve the physical plausibility of generated content. Early works such as SPNet [30] embed differentiable fluid simulations into neural networks, while Graph Network Simulators (GNS) [32] predict particle interactions using graph neural networks. Phy3DGen [43] combines 3D diffusion priors with differentiable physics layers to enforce solid mechanics constraints, and PhysDiff [47] applies biomechanical constraints to human motion synthesis in diffusion models. Similarly, Pdp [38] utilizes a diffusion policy for physics-based character animation.

Other recent video generation methods focus on learning or controlling physical representations. For example, VideoREPA [48] learns physics by aligning with foundation models, and PhysCtrl [40] focuses on physics-grounded and controllable generation. These approaches, along with works like VideoJam [6] which captures motion cues without enforcing physical laws, and PhysGen [23] which uses simulation with diffusion rendering, aim to improve physical plausibility.

NS-Diff differs by enforcing physics directly in the latent space of diffusion models. Using a lightweight reinforcement learning framework, it applies divergence and smoothness regularization for fluids and Jerk minimization for rigid bodies. This latent-space RL approach ensures physically plausible motions with minimal overhead and without relying on extensive simulations or hand-crafted constraints.

2.2. Reinforcement Learning for Video Generation

Reinforcement learning (RL) has evolved from optimizing robotic control policies [11, 12] to directly refining diffusion models [3]. In video synthesis, RL is primarily employed for cross-modal alignment [32] or human preference fine-tuning [46]. Recently, efforts have shifted toward physical grounding: PhysMaster [18] utilizes DPO to train physical encoders, while [21] leverages RL for reasoning about physical generation via timestep tokens. However, these methods [8, 9, 42, 44, 45] often overlook the explicit enforcement of physical consistency. Our method fills this gap by optimizing latent trajectories with physics-aware rewards to ensure dynamical integrity.

3. Methodology

Our framework enhances video diffusion models with physics-aware generation through three core components: (1) **Physical Attribute Detection** that identifies rigid/fluid regions directly from noisy intermediate representations, (2) **Physics-Conditioned Latent Injection** that encodes motion dynamics into the denoising process, and (3) **Re-**

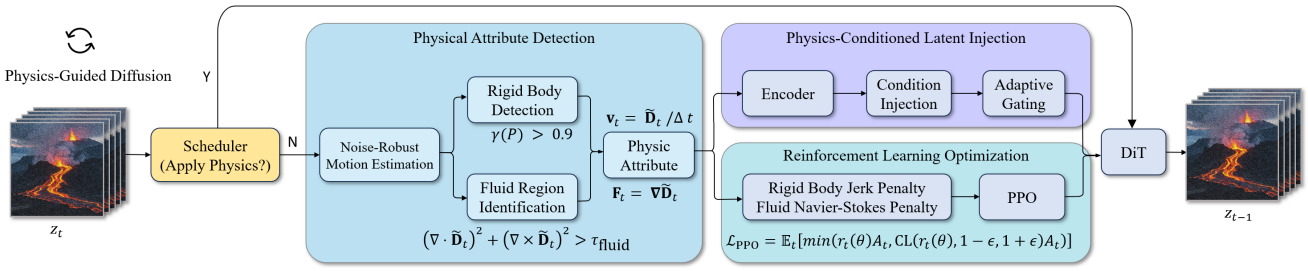


Figure 2. Our framework enhances video diffusion via three stages: (1) **Physical Attribute Detection**: noise-robust optical flow estimates motion cues and object regions; (2) **Physics-Conditioned Generation**: motion dynamics are encoded into latent representations and injected into the DiT through adaptive cross-attention; (3) **Reinforcement Learning Optimization**: policy gradients enforce rigid jerk and fluid divergence constraints. An adaptive scheduler activates physics guidance based on noise levels.

inforcement Learning Optimization that enforces physical constraints via policy gradients. As illustrated in Figure 2, we reformulate the denoising trajectory as a physics-constrained Markov Decision Process (MDP), where each denoising step optimizes both visual fidelity and physical plausibility through RL-guided latent space exploration.

3.1. Diffusion and Physical Preliminaries

Video diffusion models [15] learn to gradually denoise latent variables across T timesteps through a forward process:

$$q(\mathbf{z}_t|\mathbf{z}_{t-1}) = \mathcal{N}\left(\mathbf{z}_t; \sqrt{\alpha_t}\mathbf{z}_{t-1}, (1-\alpha_t)\mathbf{I}\right), \quad (1)$$

where α_t controls the noise schedule, $t \in [1, T]$, $\mathbf{z}_t$ is latent representation of the noisy video frame at diffusion timestep t. We reinterpret this denoising trajectory as a physics-constrained Markov Decision Process, where each step corresponds to:

- **State** s_t: the noisy latent $\mathbf{z}_t$ augmented with physical attributes (velocity, divergence, and deformation cues).
- **Action** a_t: the denoising direction $\epsilon_\theta(\mathbf{z}_t, t)$ predicted by the DiT, determining the transition $\mathbf{z}_t \to \mathbf{z}_{t-1}$.
- **Reward** $\mathcal{J}_t$: a scalar encouraging physically plausible motion, combining a rigid jerk penalty and a fluid incompressibility penalty.

3.2. Physical Attribute Detection

The key to physics-aware denoising lies in estimating motion and material attributes under high noise. A direct application of ARFlow [22] on RGB frames is infeasible, as the denoising process operates in latent space. To resolve this, we design a **hybrid latent–decoded motion estimation** scheme with the following procedure:

Global motion compensation. Before detecting local rigid or fluid regions, we first estimate global camera motion using either a **global homography** $\mathbf{H}_t$. The raw flow $\mathbf{D}_t$ is then compensated to remove camera-induced motion:

$$\mathbf{p}' = \mathbf{H}_t\mathbf{p}, \quad \mathbf{D}_t(\mathbf{p}) = \mathbf{p}' - \mathbf{p}, \quad \mathbf{p} = (x, y, 1)^\top, \quad (2)$$

Where $\mathbf{H}_t$ is a 3×3 global homography matrix modeling the estimated camera motion. Subsequent rigid/fluid detection is performed on $\mathbf{D}_t$. **Latent-to-RGB decoding for flow inference.** At each selected denoising step t, we decode the latent $\mathbf{z}_t$ into a low-resolution RGB proxy $\hat{\mathbf{x}}_t$ using the pretrained VAE decoder from the diffusion backbone: $\hat{\mathbf{x}}_t = \text{Decoder}(\mathbf{z}_t)$. This partial decoding provides spatial structure sufficient for optical flow estimation while maintaining computational efficiency. ARFlow then computes optical flow between proxies:

$$\mathbf{D}_t = \text{ARFlow}(\hat{\mathbf{x}}_{t+1}, \hat{\mathbf{x}}_t). \quad (3)$$

Latent-domain calibration. To adapt ARFlow to partially denoised, noisy frames, we fine-tune its last refinement stage using synthetic noised samples generated by the diffusion model. Specifically, we add Gaussian noise consistent with the diffusion variance schedule and train ARFlow to recover flow on these degraded pairs.

Temporal smoothing and region detection. We suppress temporal noise in the flow fields using a $(2K+1)$-frame temporal filter ($K = 2$):

$$\tilde{\mathbf{D}}_t = \frac{1}{2K+1} \sum_{k=-K}^{K} \mathbf{D}_t^k. \quad (4)$$

Rigid-Body Detection. To accurately detect planar rigid motion, we extend the local optical flow fitting to explicitly handle pure translation and affine motion, avoiding inflated residuals. For each patch $P \subset \tilde{\mathbf{D}}_t$, we solve:

$$\mathbf{D}_t(u, v) \approx \mathbf{A}\mathbf{p}_{uv} + \mathbf{b}, \quad (u, v) \in P, \quad (5)$$

where $\mathbf{p}_{uv}$ refers spatial coordinate of pixel (u, v) in homogeneous form, $\mathbf{A} \in \mathbb{R}^{2\times2}$ captures linear deformation/rotation and $\mathbf{b} \in \mathbb{R}^2$ captures pure translation. This affine formulation includes pure translation as a special case ($\mathbf{A} = \mathbf{I}$), ensuring zero residual for true planar rigid motion. The patch-wise residual is then

$$\mathcal{L}_{\text{rigid}}(P) = \sum_{(u,v)\in P} \|\tilde{\mathbf{D}}_t(u, v) - (\mathbf{A}\mathbf{p}_{uv} + \mathbf{b})\|_2^2. \quad (6)$$

A rigidity score is defined using a temperature parameter β:

$$\gamma(P) = \exp(-\beta\mathcal{L}_{\text{rigid}}(P)), \quad (7)$$

and label patches with $\gamma(P) > 0.9$ as rigid regions $\mathcal{R}_t$.

Fluid Region Identification. Regions satisfying $(\nabla \cdot \tilde{\mathbf{D}}_t)^2 + (\nabla \times \tilde{\mathbf{D}}_t)^2 > \tau_{\text{fluid}}$ are labeled as fluid $\mathcal{F}_t$, where τ_{fluid} refers threshold for identifying fluid regions, capturing locally divergent or swirling flows.

Material Embedding. For each patch $i \in \mathcal{R}_t \cup \mathcal{F}_t$, extract a patch feature $\mathbf{f}_t^{\text{patch},(i)}$ from the decoded RGB proxy using spatial pooling. Predict the material type distribution via a small MLP:

$$\mathbf{q}_t^{(i)} = \text{Softmax}\big(\text{MLP}_{\text{type}}(\mathbf{f}_t^{\text{patch},(i)})\big), \tag{8}$$

and map it to a learnable embedding: $\mathbf{e}_t^{\text{mat},(i)} = W_e \mathbf{q}_t^{(i)}$. Where W_e is a learnable linear projection matrix mapping one-hot material type distributions to embedding vectors. This embedding $\mathbf{e}_t^{\text{mat},(i)}$ is incorporated into the patch descriptor used for PCLI.

3.3. Physics-Conditioned Latent Injection

To incorporate physical cues into the diffusion process, we design a Physics-Conditioned Latent Injection (PCLI) module that encodes motion dynamics and material embeddings into the DiT's intermediate latent features. Given the smoothed optical flow $\tilde{\mathbf{D}}_t$ and material embeddings $\mathbf{e}_t^{\text{mat},(i)}$ computed in Section 3.2, we extract per-patch descriptors:

- **Velocity Field** $\mathbf{v}_t^k$: temporal derivative of the flow, $\mathbf{v}_t^{(k)} = (\tilde{\mathbf{D}}_t^{k+1} - \tilde{\mathbf{D}}_t^k)/\Delta\tau$, $\Delta\tau$ is the frame interval;
- **Deformation Gradient** $\mathbf{G}_t$: spatial Jacobian of motion, $\mathbf{G}_t = \nabla\tilde{\mathbf{D}}_t$;
- **Material Embedding** $\mathbf{e}_t^{\text{mat},(i)}$: patch-wise embedding of rigid/fluid type.

Each descriptor is resized and partitioned into 16×16 spatial patches consistent with the DiT patch tokenizer. For each patch i, the concatenated feature $\mathbf{s}_t^{(i)} = [\mathbf{v}_t^{(i)}, \mathbf{G}_t^{(i)}, \mathbf{e}_t^{\text{mat},(i)}]$ is projected to a d_p-dimensional embedding via a two-layer MLP:

$$\mathbf{p}_t^{(i)} = \text{MLP}_{\text{phys}}(\mathbf{s}_t^{(i)}), \quad \mathbf{p}_t = \{\mathbf{p}_t^{(i)}\}_{i=1}^{N_p} \in \mathbb{R}^{N_p \times d_p}. \tag{9}$$

Where N_p is the number of spatial patches in the feature map. We inject the physical latent through cross-attention:

$$\text{Attn}(\mathbf{f}_t, \mathbf{p}_t) = \text{Softmax}\left(\frac{(\mathbf{f}_t W_Q)(\mathbf{p}_t W_K)^\top}{\sqrt{d_p}}\right)(\mathbf{p}_t W_V), \tag{10}$$

where $\mathbf{f}_t$ provides queries and $\mathbf{p}_t$ serves as keys and values, W_Q, W_K, W_V are learnable projection matrices for query, key, and value transformations. The attention output is normalized and fused additively with a scaling factor $\mu = 0.1$. To dynamically adjust the influence of physics constraints along the denoising trajectory, an adaptive gate is applied:

$$g_t = \text{Sigmoid}(\text{MLP}_{\text{gate}}(t/T)), \quad \mathbf{f}_t' = \mathbf{f}_t + g_t \mu\, \text{Attn}(\mathbf{f}_t, \mathbf{p}_t). \tag{11}$$

This design ensures one-to-one spatial correspondence between physical and visual tokens, allowing motion- and material-consistent conditioning.

3.4. Reinforcement Learning Optimization

To enforce physical consistency during denoising, we formulate the diffusion process as a physics-constrained reinforcement learning problem. The goal is to maximize cumulative physical plausibility rewards across the denoising trajectory.

Policy Definition and State Transition. We parameterize the denoiser ϵ_θ as a stochastic policy $\pi_\theta(a_t|s_t)$. The action a_t, which is the predicted noise residual, is sampled from a Gaussian distribution centered at the denoiser's output $\epsilon_\theta(s_t, t)$:

$$a_t \sim \pi_\theta(a_t|s_t) = \mathcal{N}(a_t; \epsilon_\theta(s_t, t), \sigma_p^2 \mathbf{I}), \tag{12}$$

where σ_p^2 is the policy variance.

The sampled action a_t **replaces** the standard deterministic noise prediction in the DDPM reverse step, thus ensuring the state transition is policy-dependent and stochastic. This explicitly defines how the RL action a_t determines the next latent state $\mathbf{z}_{t-1}$, completing the MDP formulation. The log-probability of the sampled action a_t is:

$$\log \pi_\theta(a_t|s_t) = -\frac{1}{2\sigma_p^2}\|a_t - \epsilon_\theta(s_t, t)\|_2^2. \tag{13}$$

Trajectory Sampling. For each sample, a full denoising trajectory $\mathbf{z}_T, \ldots, \mathbf{z}_0$ is generated under the current policy. On-policy rollouts $(s_t, a_t, \mathcal{J}t, st-1)$ are collected per diffusion step and refreshed PPO epoch to ensure stability.

Rigid-Body Smoothness Regularization. For regions $\mathcal{R}_t$ (rigid areas identified in Section 3.2), we adopt the minimum-jerk prior, a computationally efficient heuristic widely used in animation and control, to enforce temporal smoothness. This constraint aims to prevent the non-physical high-frequency oscillations and temporal discontinuities often found in generated videos. We minimize the jerk $\mathbf{J}_t$, defined as the rate of change of acceleration:

$$\mathcal{L}_{\text{rigid}} = \frac{1}{|\mathcal{R}_t|} \sum_{(u,v)\in\mathcal{R}_t} \big\|\mathbf{J}_t(u,v)\big\|_2^2, \tag{14}$$

where $\mathbf{J}_t^k = (\mathbf{v}_t^{k+2} - 2\mathbf{v}_t^{k+1} + \mathbf{v}_t^k)/(\Delta\tau)^2$, $\mathbf{v}_t^k$ is the velocity field, and $\Delta\tau$ is the frame interval. While this does not model complex phenomena like collisions or friction, it serves as an effective, differentiable proxy for motion plausibility within the latent space.

Fluid Navier–Stokes Penalty. For fluid-like regions $\mathcal{F}_t$, we employ a lightweight approximation of the Navier–Stokes:

$$\mathcal{L}_{\text{fluid}} = \frac{1}{|\mathcal{F}_t^k|} \sum_{(u,v)\in\mathcal{F}_t^k} \big(\lambda_p \|\nabla(\nabla\cdot\mathbf{v}_t^k)\|_2^2 + \|\nabla\cdot\mathbf{v}_t^k\|_2^2 + \nu\|\nabla^2\mathbf{v}_t^k\|_2^2 + \eta\|\mathbf{v}_t^k\cdot\nabla\mathbf{v}_t^k\|_2^2\big) \tag{15}$$

The first term, weighted by λ_p, serves as a differentiable

proxy for the necessary pressure correction. In full fluid simulations, the pressure gradient ($-\nabla p$) is the term that actively enforces incompressibility ($\nabla \cdot \mathbf{v} = 0$). By minimizing the spatial gradient of the divergence, we encourage the flow field to converge to the divergence-free state smoothly, mimicking the spatial corrective action of pressure without requiring an expensive Poisson equation solve. The remaining terms enforce divergence-free flow, Laplacian smoothness (viscosity), and coherent flow transport (advection). The viscosity coefficient ν is adaptively predicted from the material embedding $\mathbf{e}_t^{\text{mat},(i)}$.

Reward and Value Estimation. At each denoising step, the physics-aware reward penalizes deviations from smooth, incompressible, and energy-consistent motion:

$$\mathcal{J}_t = -\lambda_1 \mathcal{L}_{\text{rigid}} - \lambda_2 \mathcal{L}_{\text{fluid}}, \tag{16}$$

where $\mathcal{L}_{\text{rigid}}$ enforces smooth rigid-body trajectories and $\mathcal{L}_{\text{fluid}}$ regularizes fluid consistency. The advantages A_t are computed via Generalized Advantage Estimation (GAE) with a value network $V_\phi(s_t)$.

Value Function. V_ϕ is a lightweight 3-layer MLP taking the same physical-augmented state as input. It is trained with mean-squared error:

$$\mathcal{L}_{\text{value}} = \tfrac{1}{2}(V_\phi(s_t) - \hat{\mathcal{J}}_t)^2. \tag{17}$$

PPO Objective. We apply Proximal Policy Optimization (PPO) [31] to update the DiT policy parameters θ:

$$\mathcal{L}_{\text{PPO}} = \mathbb{E}_t\Big[\min(r_t(\theta)A_t, \text{CL}(r_t(\theta), 1-\epsilon, 1+\epsilon)A_t)\Big], \tag{18}$$

where $r_t(\theta) = \pi_\theta(a_t|s_t)/\pi_{\text{old}}(a_t|s_t)$, A_t is the advantage function computed via Generalized Advantage Estimation, and $CL(.)$ is the clipping operation, ϵ is the clipping threshold in PPO. Gradients are backpropagated through the predicted noise ϵ_θ, enabling physics-aware refinement of the DiT. RL updates occur every $M = 4$ steps to reduce variance, alternating with value updates.

3.5. Adaptive Activation of Physics-Guided

Early denoising steps contain high noise, making motion estimation unreliable. To prevent unstable physics guidance, we introduce a **physics-activation scheduler** that modulates the strength of physics-based rewards and conditioning along the denoising trajectory.

Specifically, at step t of T total steps, the scheduler computes a weighting factor $w_t \in [0, 1]$:

$$w_t = \begin{cases} 0, & \alpha \cdot (e^{(t/T)^2} - 1) \leq \sigma \\ \alpha \cdot (t/T)^2, & \alpha \cdot (e^{(t/T)^2} - 1) > \sigma \end{cases} \tag{19}$$

where α is a scaling hyperparameter and σ decides the activate step. This weight w_t is applied multiplicatively to both: the physics-conditioned latent injection: $\mathbf{f}_t' = \mathbf{f}_t + w_t \,\text{Attn}(\mathbf{f}_t, \mathbf{p}_t)$, and the physics-based RL reward: $R_t = w_t\,(-\lambda_1 \mathcal{L}_{\text{rigid}} - \lambda_2 \mathcal{L}_{\text{fluid}})$.

Thus, early in denoising, $w_t = 0$ shuts the physics influence to avoid unstable gradients, and as denoising proceeds ($t \to T$), w_t smoothly increases toward α, gradually activating full physics guidance. This mechanism ensures stable integration of physics priors without altering the underlying diffusion noise schedule.

4. PhysVideoBench

To evaluate physical plausibility in video generation, we construct PhysVideoBench, a benchmark of 10k video clips (9k for training, 1k for testing) with detailed spatiotemporal annotations capturing diverse physical dynamics.

While datasets like UCF101 [34] and Kinetics [19] primarily assess perceptual quality, recent diagnostics such as Physics-IQ [26] have begun addressing physical common sense. However, these benchmarks often lack the granular, ground-truth motion signals required to evaluate specific conservation laws and material-specific deformations. To bridge this gap, we introduce **PhysVideoBench**, which offers curated sequences with dense physical annotations tailored to the precise dynamical constraints enforced by our model.

Train Set. For training data, we curate WebVid-10M [2] through a three-stage pipeline. First, 50,000 clips are filtered via CLIP-based keyword matching using 132 physics-centric prompts across six categories: object types, interaction verbs, material states, environments, temporal descriptors, and anomaly indicators. Second, we apply a physics saliency score combining 3D optical flow magnitudes and temporal gradients to prioritize clips with coherent motion. Finally, InstructBLIP [7] verifies physical plausibility, rejecting clips with paradoxes like interpenetrating solids or gravity-defying liquids. The resulting 9,000 training clips average 4.8 seconds at 720p, annotated with object masks, materials, and contact boundaries.

Test Set. For testing clips, to eliminate evaluation bias that would arise from relying on a single learned flow estimator, PhysVideoBench directly incorporates *ground-truth* flow where available. The 1,000 test clips are collected from optical flow datasets like Sintel [5], KITTI-2015 [25], and HD1K [20], offering real-world scenarios with dense temporal sampling and diverse materials. Each clip undergoes a multi-modal annotation pipeline:

1) **Object-level labeling** uses Grounded-SAM [29] with physics-aware prompts, combined with a rigidity classifier that computes deformation residuals to score motion rigidity. 2) **Motion verification** combines PWC-Net [35] with physics-based checks. Rigid objects are validated through Helmholtz decomposition to reject high-vorticity regions; fluid regions are tested for Navier-Stokes compliance and viscosity via shear rate estimation. 3) **Human verification** adopts a three-stage Delphi protocol with expert reviewers. Final Fleiss' $\kappa = 0.89$ indicates high agreement on physical

Table 1. Detection Accuracy under Increasing Noise

Noise Level	Rigid IoU (%)	Fluid F1 (%)	ΔJ	$\mathcal{L}_{\text{div}}$ ($\times 10^{-3}$)
0.05	98.7 ± 0.3	97.2 ± 0.5	0.08 ± 0.02	0.2 ± 0.1
0.15	95.1 ± 0.7	92.8 ± 1.1	0.18 ± 0.05	0.8 ± 0.2
0.27	89.3 ± 1.2	84.5 ± 2.1	0.33 ± 0.10	2.9 ± 0.4
0.35	71.3 ± 3.2	75.7 ± 4.1	0.58 ± 0.21	4.1 ± 1.1

Table 2. Impact of Schedulers on Guidance Steps and FVD

Scheduler	ΔJ	$\mathcal{L}_{\text{div}}$ ($\times 10^{-3}$)	Guidance Steps	FVD ↓
Linear	0.67 ± 0.15	5.9 ± 1.2	54	207
Ours	0.33 ± 0.10	2.9 ± 0.4	84	183

plausibility, with 98% consistency between automated and manual annotations after resolving edge cases.

5. Experiments

5.1. Dataset and Metrics

To verify the effectiveness of our model, we conduct experiments on both our proposed PhysVideoBench and standard benchmarks, including UCF-101 [34] (13,320 real-world human action videos) and WebVid-10M [1] (10M captioned internet videos). We evaluate our approach from two perspectives: physical plausibility and visual quality.

Physical Metrics. 1) *Rigid Body Accuracy*: Intersection-over-Union (IoU) between detected rigid regions and annotated rigid masks. 2) *Fluid F1 Score*: Harmonic mean of precision and recall for fluid-region detection. 3) *Jerk Consistency (ΔJ)*: Measures the temporal smoothness of rigid-body motion by computing the mean squared magnitude of the third-order temporal derivative of centroid trajectories, 4) *Fluid Divergence Error ($\mathcal{L}_{div}$)*: L_2 norm of divergence computed from GT flow or depth-aligned motion proxies. Note that all physical quantities are computed directly from the annotated spatiotemporal trajectories to avoid potential bias from any learned optical-flow estimator.

Visual Metrics. 1) *VBench* [17]: Comprehensive video quality metrics; we follow VideoJam [6] for evaluation of appearance and motion quality. 2) *Fréchet Video Distance* (FVD) [39]: Distributional similarity between real and generated videos. 3) *Frame Consistency* [10]: Average cosine similarity between CLIP image embeddings of consecutive frames, reflecting temporal coherence.

5.2. Implementation Details

Our framework is built on the pre-trained DiT-based architecture [28]. The physics-aware components include ARFlow for optical flow estimation, integrated via cross-attention layers with a latent injection dimension of $d_p = 128$. Motion and material attributes are computed on 16×16 patches with a temporal smoothing window of $K = 2$ (i.e., 5-frame averaging). Rigid-body smoothness (jerk) regularization uses $\Delta\tau$ equal to the inter-frame interval, and fluid divergence–smoothness regularization employs coefficients $\eta = 0.05$ (advection) and $\lambda_p = 0.01$. The rigid/fluid loss weights are set to $\lambda_1 = 1.0$ and $\lambda_2 = 1.0$ unless specified otherwise.

Table 3. Video Generation Performance on PhysVideoBench

Method	ΔJ	$\mathcal{L}_{\text{div}}$	Appear. ↑	Motion ↑
VideoJam [6]	0.74	4.7	71.6	90.1
PhysGen [23]	0.63	3.5	72.8	91.0
OpenSora2-1B [28]	0.72	4.6	70.4	89.2
OpenSora2-11B [28]	0.69	4.0	71.9	90.3
Wan2.1 [36]	0.67	3.7	72.6	90.6
NS-Diff-DiT 1B	0.33	2.9	73.1	92.4
NS-Diff-DiT 11B	**0.25**	**2.4**	**74.4**	**93.7**

For reinforcement learning, the physics-constrained PPO optimization uses a Gaussian policy with standard deviation $\sigma_p = 0.1$, clip parameter $\epsilon = 0.2$, and GAE $\lambda = 0.95$. On-policy rollouts are collected over 32 diffusion steps with mini-batches of 512 transitions per PPO epoch, and 4 PPO epochs are performed per update. Policy and value function updates alternate every 4 diffusion steps, while the value function is trained with MSE against discounted returns ($\gamma = 0.99$). The standard diffusion loss is retained during RL fine-tuning to stabilize training.

Unless otherwise specified, all ablation studies are conducted on the PhysVideoBench dataset using the 1B DiT model. Experiments run on fewer than 8× NVIDIA A40 GPUs for approximately 4 days. The lightweight MLP policy and latent-level physics computation introduce minimal training overhead.

5.3. Analysis of Physics Estimation from Noise

We first evaluate the reliability of our physics estimator when operating on noisy intermediate latents (z_t) at various noise levels. The experiment, summarized in Table 1, involves decoding z_t into proxy images and measuring key physical metrics. The results clearly show that estimation fidelity degrades significantly as noise increases. For instance, as the noise level increases from 0.05 to 0.35, Rigid IoU drops from 98.7% to 71.3%, while the fluid divergence error ($\mathcal{L}_{\text{div}}$) increases by an order of magnitude.

This finding confirms that applying guidance when noise is high (i.e., in the early denoising stages) would propagate erroneous gradients and destabilize optimization. Therefore, our adaptive scheduler (eq. (19)) is designed to activate physics guidance only during timesteps where the estimates are reliable (i.e., at lower noise levels), ensuring stable and effective optimization. The superiority of this strategy is validated in the ablation study in Table 2.

5.4. Comparison with the State-of-the-art

We evaluate unconditional video generation performance on three benchmarks: **UCF-101**, **WebVid**, and **PhysVideoBench**. For fair comparison, all methods gener-

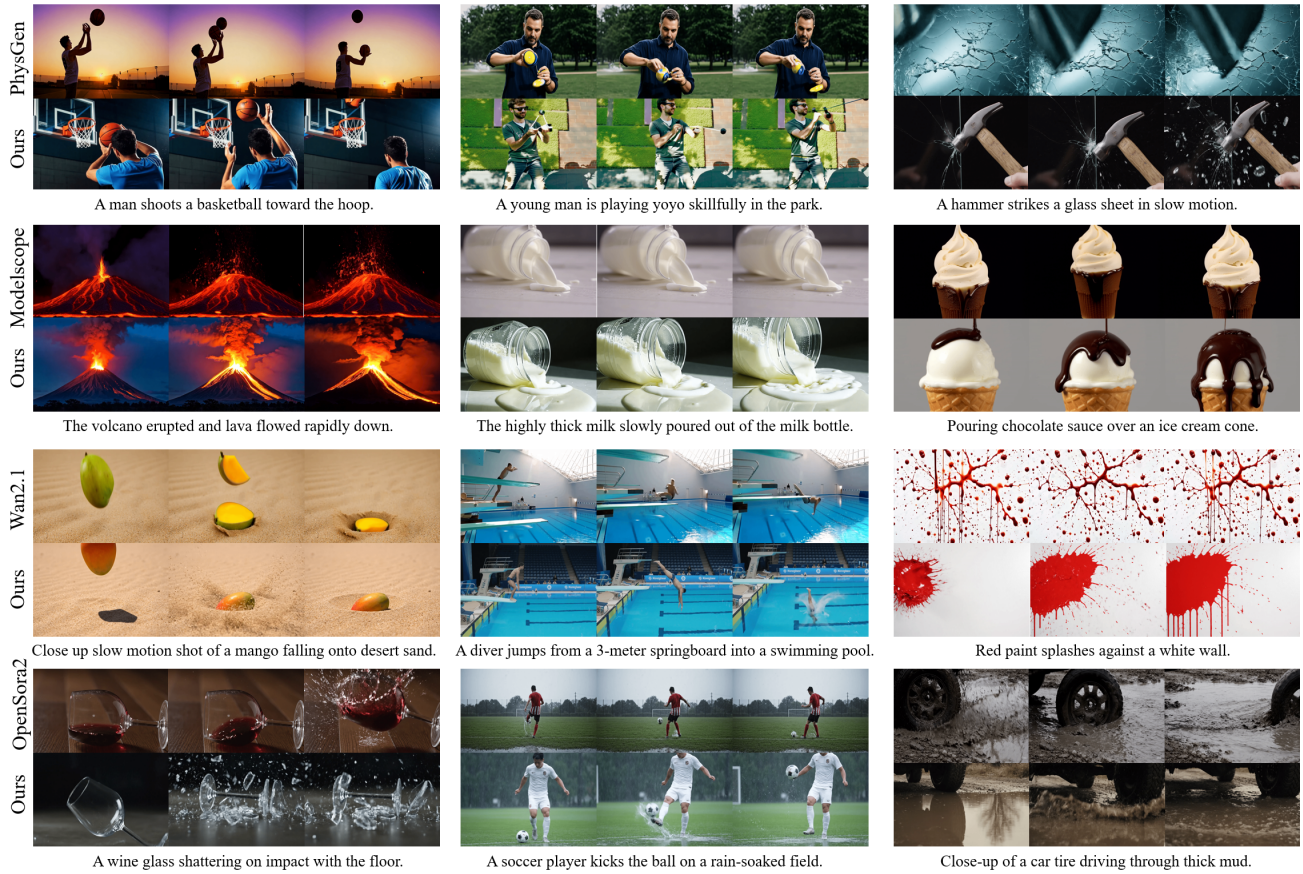


Figure 3. Visual comparison of NS-Diff, ModelScope, PhysGen, Wan2.1, and OpenSora2. Our method produces more realistic rigid and fluid motion, reduces unphysical artifacts (object appearance/disappearance, unnatural splitting/merging), and improves frame consistency.

Table 4. Video Generation Performance on UCF-101.

Method	Year	FVD ↓	Frame Cons. ↑
VDM [15]	NeurIPS 2022	295	0.82
CogVideo [16]	ICLR 2023	305	0.88
OpenSora2-1B [28]	Arxiv 2025	134	0.88
OpenSora2-11B [28]	Arxiv 2025	110	0.91
NS-Diff DiT 1B	Ours	106	0.94
NS-Diff DiT 11B	**Ours**	**85**	**0.95**

Table 5. WebVid-10M Text-to-Video Generation Comparison.

Method	Year	FVD ↓	CLIPSIM ↑
LVDM [13]	Arxiv 2022	456	0.28
ModelScope [24]	CVPR 2023	414	0.30
VideoFactory [41]	IJCV 2025	322	0.31
OpenSora2 [28]	Arxiv 2025	297	0.32
NS-Diff	Ours	**275**	**0.34**

ate 2,048 video clips (16 frames, 256×256 resolution) without text prompts or class labels. FVD metrics follow the standard protocol of prior work [15], using an I3D model pretrained on Kinetics-400. Crucially, none of the methods (including ours) utilize physics annotations (e.g., fluid/rigid body region, velocity fields) during training—our physics constraints derive solely from real-time differentiable simulations of intermediate latent frames, ensuring comparability with the compared methods.

On PhysVideoBench, NS-Diff achieves the best performance across all metrics. Incorporating physics priors via latent injection and Jerk/divergence losses significantly improves motion realism, particularly for rigid bodies and fluid regions. Compared to PhysGen, which simulates rigid-body dynamics given user forces, NS-Diff generalizes without requiring predefined forces or simulations, while maintaining higher fidelity and lower divergence errors.

To evaluate how effectively our physics-guided learning framework generalizes to open-world scenarios beyond constrained benchmarks, we conducted text-to-video synthesis on the WebVid-10M dataset, testing whether physical constraints (rigid/fluid dynamics) maintain plausibility for unseen textual descriptions involving novel object interactions and environments. Following VideoFactory's [41] protocol, we evaluated a large set of unseen videos using FVD for motion quality and CLIPSIM for text-video alignment. Our model outperforms VideoFactory on both metrics. Improvements are especially notable in physics-intensive scenes, where physical regularization reduces im-

Table 6. Component Ablation Study on PhysVideoBench.

Configuration	Appear. ↑	Motion ↑	ΔJ	$\mathcal{L}_{\text{div}}$ ($\times 10^{-3}$)
Full Model (Physics + PPO)	**73.1**	**92.4**	**0.33**	**2.9**
Physics Losses w/o RL	71.5	90.1	0.58	4.7
w/o Condition Injection	70.0	88.1	0.82	6.9
w/o Physics Constraints	71.0	89.0	1.21	8.4
w/o Adaptive Scheduler	72.2	91.0	0.67	4.1
w/o $\mathcal{L}_{\text{fluid}}$	71.7	90.5	0.49	10.4
w/o $\mathcal{L}_{\text{rigid}}$	70.8	89.6	1.33	3.5

Table 7. Hyperparameter Sensitivity Analysis.

σ	η	Activate Step t	FVD ↓	ΔJ	$\mathcal{L}_{\text{div}}$ ($\times 10^{-3}$)
0.25	0.3	22	197	0.54	4.6
0.27	**0.5**	**16**	**183**	**0.33**	**2.9**
0.30	0.7	10	215	0.61	6.3
0.27	0.2	16	191	0.39	3.4
0.27	0.5	16	189	0.36	3.1

plausible motions while preserving semantic coherence. This shows that learned physical priors generalize well from structured physical datasets to open-domain video generation, balancing generative diversity and dynamic realism.

Visualization results in Fig. 3 demonstrate that our method not only improves rigid-body and fluid dynamics fidelity but also **effectively mitigates common unphysical artifacts** such as sudden object appearance or disappearance, unnatural splitting or merging of entities. These behaviors are inconsistent with rigid body and fluid mechanics and are discouraged during reinforcement learning.

5.5. Ablation Studies

Component Analysis. We conduct ablations on PhysVideoBench to evaluate each module (Table 6). A baseline *Physics Losses w/o RL* applies the same physics objectives without PPO optimization. While it encourages local physical consistency, it lacks temporal stability and shows higher divergence, indicating PPO enhances dynamic coherence beyond direct gradients.

Removing condition injection degrades both visual and physical quality, while dropping physics constraints leads to discontinuous motion. The adaptive scheduler stabilizes denoising under high noise, and removing either $\mathcal{L}_{\text{fluid}}$ or $\mathcal{L}_{\text{rigid}}$ increases deformation or fluid instability. These results show that PPO-based optimization and physics-guided losses enforce physically consistent video generation.

Hyperparameter Sensitivity. Table 7 analyzes the effect of noise threshold σ and advection weight η. The best configuration ($\sigma = 0.27, \eta = 0.5$) achieves the lowest FVD and divergence, balancing physical stability and generative flexibility. Smaller η weakens advection consistency, causing unstable flow near object boundaries, while excessively large η enforces overly rigid motion.

Flow Backbone. We test three backbones (ARFlow, RAFT [37], PWC-Net [35]) in two domains: *Decoded* (latent decoded to low-res RGB) and *Latent* (direct latent estimation). Decoded-domain estimation consistently outperforms latent-domain estimation. ARFlow achieves the best trade-off between accuracy and efficiency, and the minor $\sim$8% overhead is justified by gains in ΔJ and $\mathcal{L}_{\text{div}}$.

Table 8. Ablation on flow backbone and estimation domain.

Backbone	**Domain**	ΔJ ↓	$\mathcal{L}_{\text{div}}$ ($\times 10^{-3}$)	FVD ↓	Overhead (%)
ARFlow	Decoded	**0.33**	**2.9**	**183**	8
RAFT [37]	Decoded	0.42	3.6	190	10
PWC-Net [35]	Decoded	0.48	4.2	198	6
ARFlow	Latent	0.57	5.3	205	4
RAFT [37]	Latent	0.61	5.8	211	5
PWC-Net [35]	Latent	0.67	6.4	218	4

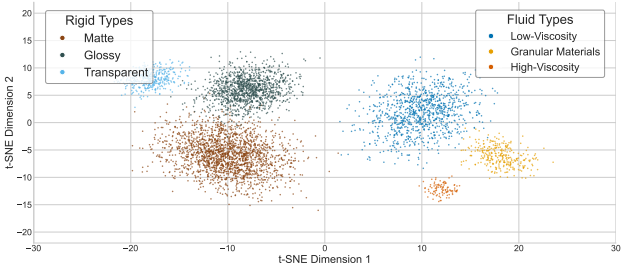


Figure 4. t-SNE visualization of material embeddings.

Material Embedding Analysis. t-SNE visualization (Fig. 4) reveals that our model autonomously learns fine-grained physical categories beyond simple macro-clusters. Specifically, fluids are sub-partitioned into **low/high-viscosity** and **granular** groups, while rigid bodies cluster by **matte, glossy, and transparent** attributes. This self-organized hierarchy suggests a deep physical understanding, enabling more realistic video synthesis.

6. Conclusion

We propose NS-Diff, a physics-guided video generation framework that integrates reinforcement learning with diffusion models to enforce physical consistency without labeled supervision. By formulating rigid body dynamics (minimum jerk principle) and fluid constraints (simplified Navier–Stokes with soft pressure projection) as differentiable reward functions, our method significantly improves physical accuracy and reduces a broad range of non-physical artifacts.

Beyond quantitative gains, NS-Diff demonstrates that latent-level physics priors can effectively guide generative models, suggesting a general strategy for embedding domain knowledge without explicit supervision. The approach also highlights how simplified, differentiable physical approximations can strike a practical balance between realism and computational efficiency.

Acknowledgements

This work was supported by the grants from the National Natural Science Foundation of China (62132001, 62525201, 62432001) and Beijing Natural Science Foundation (L247006).

References

[1] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1708–1718, 2021. 6

[2] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *IEEE International Conference on Computer Vision*, 2021. 5

[3] Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning. In *The Twelfth International Conference on Learning Representations*, 2024. 2

[4] Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, Clarence Ng, Ricky Wang, and Aditya Ramesh. Video generation models as world simulators. 2024. 1, 2

[5] Daniel J Butler, Jonas Wulff, Garrett B Stanley, and Michael J Black. A naturalistic open source movie for optical flow evaluation. In *Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part VI 12*, pages 611–625. Springer, 2012. 5

[6] Hila Chefer, Uriel Singer, Amit Zohar, Yuval Kirstain, Adam Polyak, Yaniv Taigman, Lior Wolf, and Shelly Sheynin. Videojam: Joint appearance-motion representations for enhanced motion generation in video models. In *International Conference on Machine Learning*, pages 1–14. PMLR, 2025. 1, 2, 6

[7] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning, 2023. 5

[8] Zijun Deng, Xiangteng He, and Yuxin Peng. Efficiency-optimized video diffusion models. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 7295–7303, 2023. 2

[9] Zijun Deng, Xiangteng He, Yuxin Peng, Xiongwei Zhu, and Lele Cheng. Mv-diffusion: Motion-aware video diffusion model. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 7255–7263, 2023. 2

[10] Patrick Esser, Johnathan Chiu, Parmida Atighehchian, Jonathan Granskog, and Anastasis Germanidis. Structure and content-guided video synthesis with diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7346–7356, 2023. 6

[11] Chelsea Finn, Ian Goodfellow, and Sergey Levine. Unsupervised learning for physical interaction through video prediction. *Advances in neural information processing systems*, 29: 64–72, 2016. 2

[12] Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. *arXiv preprint arXiv:1912.01603*, 2019. 2

[13] Yingqing He, Tianyu Yang, Yong Zhang, Ying Shan, and Qifeng Chen. Latent video diffusion models for high-fidelity long video generation. *arXiv preprint arXiv:2211.13221*, 2022. 7

[14] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020. 1

[15] 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. 3, 7

[16] Wenyi Hong, Ming Ding, Wendi Zheng, Xinghan Liu, and Jie Tang. Cogvideo: Large-scale pretraining for text-to-video generation via transformers. In *International Conference on Learning Representations*, pages 1–13, 2023. 7

[17] 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. 6

[18] Sihui Ji, Xi Chen, Xin Tao, Pengfei Wan, and Hengshuang Zhao. Physmaster: Mastering physical representation for video generation via reinforcement learning. *arXiv preprint arXiv:2510.13809*, 2025. 2

[19] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017. 5

[20] Daniel Kondermann, Rahul Nair, Katrin Honauer, Karsten Krispin, Jonas Andrulis, Alexander Brock, Burkhard Gussefeld, Mohsen Rahimimoghaddam, Sabine Hofmann, Claus Brenner, et al. The hci benchmark suite: Stereo and flow ground truth with uncertainties for urban autonomous driving. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 19–28, 2016. 5

[21] Wang Lin, Liyu Jia, Wentao Hu, Kaihang Pan, Zhongqi Yue, Wei Zhao, Jingyuan Chen, Fei Wu, and Hanwang Zhang. Reasoning physical video generation with diffusion timestep tokens via reinforcement learning. *arXiv preprint arXiv:2504.15932*, 2025. 2

[22] Liang Liu, Jiangning Zhang, Ruifei He, Yong Liu, Yabiao Wang, Ying Tai, Donghao Luo, Chengjie Wang, Jilin Li, and Feiyue Huang. Learning by analogy: Reliable supervision from transformations for unsupervised optical flow estimation. In *IEEE Conference on Computer Vision and Pattern Recognition(CVPR)*, 2020. 3

[23] 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. Springer, 2024. 2, 6

[24] Zhengxiong Luo, Dayou Chen, Yingya Zhang, Yan Huang, Liang Wang, Yujun Shen, Deli Zhao, Jingren Zhou, and Tieniu Tan. Videofusion: Decomposed diffusion models for high-quality video generation. In *Proceedings of*

the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 10209–10218, 2023. 7

[25] Moritz Menze, Christian Heipke, and Andreas Geiger. Joint 3d estimation of vehicles and scene flow. *ISPRS annals of the photogrammetry, remote sensing and spatial information sciences*, 2:427–434, 2015. 5

[26] Saman Motamed, Laura Culp, Kevin Swersky, Priyank Jaini, and Robert Geirhos. Do generative video models understand physical principles?, 2025. 5

[27] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023. 1

[28] Xiangyu Peng, Zangwei Zheng, Chenhui Shen, Tom Young, Xinying Guo, Binluo Wang, Hang Xu, Hongxin Liu, Mingyan Jiang, Wenjun Li, et al. Open-sora 2.0: Training a commercial-level video generation model in 200 k. *arXiv preprint arXiv:2503.09642*, 2025. 6, 7

[29] Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, et al. Grounded sam: Assembling open-world models for diverse visual tasks. *arXiv preprint arXiv:2401.14159*, 2024. 5

[30] Connor Schenck and Dieter Fox. Spnets: Differentiable fluid dynamics for deep neural networks. In *Conference on Robot Learning*, pages 317–335. PMLR, 2018. 2

[31] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. 5

[32] Younggyo Seo, Kimin Lee, Stephen L James, and Pieter Abbeel. Reinforcement learning with action-free pre-training from videos. In *International Conference on Machine Learning*, pages 19561–19579. PMLR, 2022. 2

[33] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 1

[34] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012. 5, 6

[35] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8934–8943, 2018. 5, 8

[36] Wan Team. Wan: Open and advanced large-scale video generative models. 2025. 6

[37] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *European conference on computer vision*, pages 402–419. Springer, 2020. 8

[38] Takara Everest Truong, Michael Piseno, Zhaoming Xie, and Karen Liu. Pdp: Physics-based character animation via diffusion policy. In *SIGGRAPH Asia 2024 Conference Papers*, pages 1–10, 2024. 2

[39] Thomas Unterthiner, Sjoerd van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric & challenges. *arXiv preprint arXiv:1812.01717*, 2018. 6

[40] Chen Wang, Chuhao Chen, Yiming Huang, Zhiyang Dou, Yuan Liu, Jiatao Gu, and Lingjie Liu. Physctrl: Generative physics for controllable and physics-grounded video generation. *NeurIPS*, 2025. 2

[41] Wenjing Wang, Huan Yang, Zixi Tuo, Huiguo He, Junchen Zhu, Jianlong Fu, and Jiaying Liu. Swap attention in spatiotemporal diffusions for text-to-video generation. *International Journal of Computer Vision*, pages 1–19, 2025. 7

[42] Fan Xie, Dan Zeng, Qiaomu Shen, and Bo Tang. A comprehensive survey on text-to-video generation. *Chinese Journal of Electronics*, 34(4):1009–1036, 2025. 2

[43] Qingshan Xu, Jiao Liu, Melvin Wong, Caishun Chen, and Yew-Soon Ong. Precise-physics driven text-to-3d generation. *arXiv preprint arXiv:2403.12438*, 2024. 1, 2

[44] Zhiwen Yang and Yuxin Peng. Gala-2.5 d: Global-local alignment with 2.5 d semantic guidance for camera-based 3d semantic scene completion in autonomous driving. *Chinese Journal of Electronics*, 2025. 2

[45] Zhaoda Ye, Xiangteng He, and Yuxin Peng. Unsupervised cross-media hashing learning via knowledge graph. *Chinese Journal of Electronics*, 31(6):1081–1091, 2022. 2

[46] Hangjie Yuan, Shiwei Zhang, Xiang Wang, Yujie Wei, Tao Feng, Yining Pan, Yingya Zhang, Ziwei Liu, Samuel Albanie, and Dong Ni. Instructvideo: Instructing video diffusion models with human feedback. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6463–6474, 2024. 2

[47] Ye Yuan, Jiaming Song, Umar Iqbal, Arash Vahdat, and Jan Kautz. Physdiff: Physics-guided human motion diffusion model. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 16010–16021, 2023. 1, 2

[48] 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. *NeurIPS*, 2025. 2