PhysRVG: Physics-Aware Unified Reinforcement Learning for Video Generative Models

Qiyuan Zhang[1,2*], Biao Gong[2†], Shuai Tan[2], Zheng Zhang[2], Yujun Shen[2], Xing Zhu[2], Yuyuan Li[1], Kelu Yao[3], Chunhua Shen[1], and Changqing Zou[1,3‡]

[1]Zhejiang University, China [2]Ant Group, China [3]Zhejiang Lab, China
{zhangqiyuan, changqing.zou}@zju.edu.cn

Abstract. Physical principles are fundamental to realistic visual simulation, but remain a significant oversight in video generation. This gap highlights a critical limitation in rendering rigid body motion, a core tenet of classical mechanics. While computer graphics and physics-based simulators can easily model such dynamics using Newton formulas, modern video generative models discard the concept of object rigidity during pixel-level global denoising. Existing methods attempt to tackle this problem through physical data augmentation, dynamics pre-simulation, or reinforcement learning with VLM ratings, but none of these approaches accurately reflect physical principles or enable the model to internalize physical knowledge. Motivated by these considerations, we introduce reinforcement learning with physically verifiable rewards. We design a quantitatively verifiable metric that combines Trajectory Offset and Collision Detection, which can accurately capture rigid-body motion states and assess the quality of generated samples. Subsequently, we extend this paradigm to a unified post-training framework, termed **Mimicry-Discovery Cycle**, which enables stable training on out-of-distribution scenarios while improving overall model performance. To validate our approach, we construct new benchmark `PhysRVGBench` and perform extensive qualitative and quantitative experiments to thoroughly assess its effectiveness. The code and demo can be found at `https://lucaria-academy.github.io/PhysRVG/`.

Keywords: Physically Verifiable Rewards · Mimicry-Discovery Cycle · Rigid Body Motion

1 Introduction

Physical principles form the foundation of realistic visual simulation, governing how objects move, interact, and respond to forces in the real world [3,18,30,37]. From rigid body motion to the deformation of soft materials, physical laws establish the continuity and coherence that make dynamic scenes appear physically plausible. In traditional computer graphics and simulation pipelines [21,56], such principles are explicitly encoded through Newtonian mechanics and numerical solvers [36,38], ensuring that visual outcomes remain physically consistent.

*Work done during internship at Ant Group.
†Project lead. ‡Corresponding author.

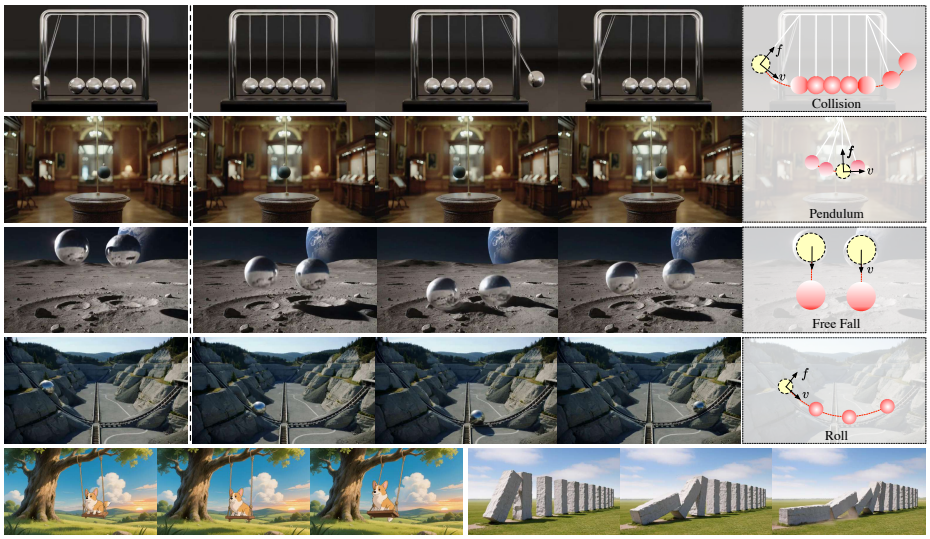

Fig. 1: Samples generated by `PhysRVG`. Our model produces videos with physically plausible rigid body dynamics. *Rows1–4* display four fundamental types of motion addressed in our work, *row5* validates the generalization in out-of-distribution scenarios.

However, recent advances in transformer-based video generation have shifted the focus toward data-driven synthesis, where realism emerges statistically from large-scale training rather than from explicit physical modeling [1, 10, 11, 35, 49]. While this paradigm has enabled remarkable progress in semantic understanding and visual fidelity [27, 37], it remains inherently limited in representing the underlying dynamics of physical real motion. The absence of embedded physical constraints leads to inconsistencies such as unstable trajectories [22], implausible collisions [1], and a lack of temporal coherence [4], especially when modeling rigid body motion governed by classical mechanics [23, 32].

Recently, many works have attempted to address this problem. We summarize the core ideas across methods in Fig. 2. Yet, existing approaches still suffer from inherent limitations: they cannot accurately reflect physical principles or enable the model to internalize physical knowledge. We therefore ask whether it is possible to design a method that can accurately capture the motion process and faithfully reflect physical rules, providing precise optimization signals to guide the model toward physically plausible trajectories.

Motivated by these insights, we introduce `PhysRVG`, a reinforcement learning with physically verifiable rewards paradigm. In this work, we focus on *Rigid Body Motion* [32] as the core representation of physical behavior. Most rigid body motions fall within the scope of classical mechanics, where object states can be precisely modeled as the temporal variation of coordinates. To quantitatively verify and evaluate motion plausibility in videos, we introduce a physics-grounded reward function that integrates Trajectory Offset and Collision Detection. The former reflects motion accuracy, while the latter encourages the model to focus

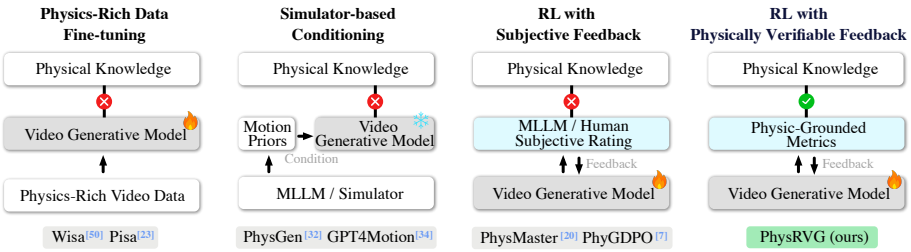

Fig. 2: Core Idea Comparison of Different Methods. Four distinct methodologies for enhancing physical realism are illustrated from left to right. (1) Physics-Rich Data Fine-tuning fundamentally learns statistical similarities in pixel distributions, failing to acquire a genuine understanding of underlying physical dynamics. (2) Simulator-based Conditioning leverages MLLMs or physics engines to generate motion priors (*e.g.,* optical flow) as spatial-temporal controls, which prevents the generation model from internalizing intrinsic physical knowledge. (3) RL with Subjective Feedback uses human or MLLM preferences for post-training. Unfortunately, MLLMs and human subjective rating lack the precision to accurately evaluate the physical correctness of samples. (4) In contrast, our proposed method utilizes physically verifiable metrics to precisely quantify the physical realism of each sample, providing the model with precise, physics-grounded optimization guidance, enabling it to truly internalize physical laws.

on challenging collision events. However, simply combining this reward function with RL algorithm resulted in highly unstable training and suboptimal performance. We attribute this to the model's inability to generate reasonable samples in out-of-distribution scenarios within limited attempts, thereby preventing RL from providing effective optimization signals in challenging scenarios. To address this, we extend `PhysRVG` into a unified post-training framework through the proposed **Mimicry-Discovery Cycle** (`MDcycle`).

This method dynamically alternates between the *"Mimicry"* and *"Discovery"* phases based on rollout quality. During the *"Mimicry"* phase, the model captures visual patterns from the data and mitigates unreliable reward signals in standard reinforcement learning. Conversely, the *"Discovery"* phase enables the model to progressively internalize physical rules, facilitating a smooth transition from data-driven learning to physics-consistent generation.

For comprehensive training and evaluation of `PhysRVG`, we construct PhysRVGBench, a benchmark comprising 700 videos collected from game footage, online sources, and in-house recordings. These videos cover four types of rigid body motion: collision, pendulum, free fall, and rolling. For each generated video, we compute two key evaluation metrics, Intersection over Union (IoU) [43] and Trajectory Offset (TO), which provide precise, quantitative measures of physical plausibility in rigid body dynamics. Experimental results in Fig. 1 and Sec. 4 validate the effectiveness of our approach across a wide range of physical and visual settings. Our key contributions include:

- We are the first to introduce reinforcement learning with physically verifiable rewards into video generation. We propose a physics-grounded reward that can accurately evaluate the plausibility of rigid body motions.
- We propose a new post-training algorithm `MDcycle`, which enables stable training on out-of-distribution and challenging scenarios while improving overall model performance.
- We construct a new benchmark, PhysRVGBench. Extensive quantitative and qualitative experiments demonstrate that PhysRVG outperforms state-of-the-art video generation models in both visual realism and physical accuracy.

2 Related Work

2.1 Physics-Aware Video Generation.

While modern video generative models [11, 15, 22, 39, 46, 47, 53, 60] can render highly realistic frames, their grasp of physical laws remains insufficient [6, 19, 27, 54]. Methods to enhance physical fidelity in video generation can be categorized by how they inject knowledge [2,7–9,16,25,26,28,48,63]. The first class conditions generation on explicit motion priors. PhysGen [32] derives motion sequences from rigid body dynamics simulation, GPT4Motion [34] leverages GPT-4o planning and Blender simulation to obtain edge maps and depth maps, NewtonGen [62] and PhysAnimator [55] use optical flow as input guidance. These methods depend heavily on the quality and coverage of the simulator. The second approach expands training data with more physics related examples. Wisa [52] builds a dataset containing multi-level basic physics knowledge and finetune the model using MoE. Pisa [23] collects free fall videos and applies post-training to teach the model specific physical behaviors. However, such methods inherently confine the model's capabilities to the motion modes represented in the curated datasets. The third class learns from reward feedback. PhyT2V [57] employs a MLLM to iteratively evaluate generations and refine textual prompts, while PhysMaster [20] uses human feedback to rank samples and applies DPO [41] for preference optimization. These pipelines depend on subjective feedback, while our method leverages Trajectory Offset (TO) and Collision Detection, which can quantitatively and accurately assess the quality of generated samples.

2.2 Reinforcement Learning for Generative Model.

Reinforcement learning has achieved strong success in large language models [12, 40,59]. RLHF [14] uses human ranking and preferences as feedback and optimizes the model with Proximal Policy Optimization(PPO) [44]. DeepSeek -R1 [13] uses verifiable outcomes as feedback and applies Group Relative Policy Optimization(GRPO) [45] to greatly improve reasoning. Compared to PPO, GRPO is more efficient because it does not require training a separate value model. Motivated by these successes, DDPO [5] brings reinforcement learning into diffusion-based generative models and formulates the denoising process as a Markov Decision Process(MDP). Recently, Flow-GRPO [31] and DanceGRPO [58] push

GRPO into flow matching models by converting ODE sampling into equivalent SDE sampling and adding tunable exploration noise. For faster training, MixGRPO [24] proposes a hybrid ODE-SDE strategy. The model reaches competitive results with very few optimization steps and a short training time. TempFlowGRPO [17] and G^{2}RPO [64] leverage the intrinsic temporal structure of flow-based models and address uneven credit assignment under sparse rewards. Existing methods are largely confined to the image domain and focus on low level attributes such as style changes. In contrast, our method extends to high level semantic attributes in videos, such as object motion, which are more complex and harder to optimize.

3 Methodology

In this section, we first present the necessary background on reinforcement learning and flow matching in Sec. 3.1. In Sec. 3.2, we define our task and articulate its relationship with physical knowledge. Subsequently, Sec. 3.3 details the design of our reward function. We then present `MDcycle`, a unified post-training method in Sec. 3.4. Finally, Sec. 3.5 outlines the overall training pipeline. The overall framework of `PhysRVG` is illustrated in Fig. 3.

3.1 Preliminary

Flow Matching. Flow Matching [29,33] treats generation as a denoising process from noise to data. Assume x_0 is sampled from real data and x_1 is sampled from Gaussian noise distribution. The intermediate sample x_t is a linear interpolation of x_0 and x_1, defined as: $x_t = (1-t)x_0 + tx_1$. A generative model is trained to predict the velocity field $v = x_1 - x_0$. The optimization objective for this process can be formulated as:

$$L(\theta) = \mathbb{E}_{t,x_0\sim X_0,x_1\sim X_1}[\|v - v_\theta(x_t,t)\|^2]. \tag{1}$$

Reinforcement Learning. According to Flow-GRPO [31] and DanceGRPO [58], the ODE governing the Flow Matching denoising trajectory can be convert to SDE while preserving the same marginal distributions, the SDE denoising process can be formulated as:

$$\mathrm{d}x_t = \left[v_t + \frac{\sigma_t^2}{2t}\left(x_t + (1-t)v_t\right)\right]\mathrm{d}t + \sigma_t\,\mathrm{d}w \tag{2}$$

where v_t is the predicted velocity, σ_t is a hyperparameter controlling the strength of stochastic noise (i.e., the intensity of exploration in RL), and w represents standard Brownian motion.

GRPO estimates the advantage value for each sample by comparing it against other samples within the same group. Given a group of G samples $\{x_0^i\}_{i=1}^G$ generated from the same condition c, the advantage corresponding to the i-th sample is formulated as:

$$A_t^i = \frac{r(x_0^i,c) - \mathrm{mean}(\{r(x_0^j,c)\}_{j=1}^G)}{\mathrm{std}(\{r(x_0^j,c)\}_{j=1}^G)} \tag{3}$$

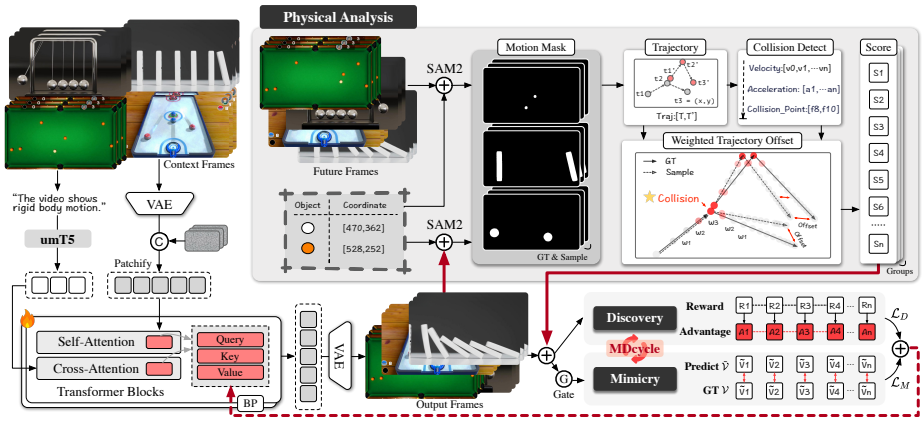

Fig. 3: The framework of PhysRVG. Given a text prompt and context frames, the model generates future video frames. For both the groundtruth and sampled frames, we derive motion masks M by prompting SAM2 [42] with object coordinates p_1 from the first frame, which are annotated during data preprocessing. We then compute object trajectories P and perform collision detection. The trajectory offset O between the sampled and groundtruth trajectories is calculated and reweighted by the collision signal w_t to yield a weighted trajectory offset O_c, which serves as the per-sample score.

The GRPO algorithm then optimizes the policy model by maximizing the following objective:

$$J(\theta) = \frac{1}{G}\sum_{i=1}^{G}\frac{1}{T}\sum_{t=0}^{T-1}\Big(\min\big(r_t^i(\theta)\,\hat{A}_t^i,\\ \text{clip}(r_t^i(\theta),\,1-\epsilon,\,1+\epsilon)\,\hat{A}_t^i - D_{KL}(\pi_\theta \,\|\, \pi_{\text{ref}})\big)\Big), \tag{4}$$

where $r_t^i(\theta) = \frac{p_\theta(x_{t-1}^i|x_t^i,\boldsymbol{c})}{p_{\theta_{old}}(x_{t-1}^i|x_t^i,\boldsymbol{c})}$ serves as an importance sampling ratio, correcting for the bias of evaluating the current policy using data generated by an older one. The D_{KL} term acts as a regularizer to promote training stability.

3.2 Physics-Grounded Task Definition

Physical laws provide a natural foundation for defining verifiable tasks in video generation. Among them, rigid body motion serves as a fundamental and well-defined phenomenon that enables quantitative evaluation. It exhibits two intrinsic properties: (1) **Observability**: the motion of a rigid body can be precisely represented through coordinate transformations, and its position is directly measurable from video frames; (2) **Determinism**: given the initial position and velocity, the subsequent trajectory of a rigid body is uniquely determined by Newtonian mechanics. These properties make rigid body motion suitable for verifying the physical plausibility of generated videos. In this work, we consider four representative types of motion: collision, pendulum, free fall, and rolling.

Specifically, our task can be formulated as simulating the rigid body dynamics by generating future frames based on the observed initial frames. Given the initial T_{obs} frames $\{I_1, I_2, \ldots, I_{T_{\text{obs}}}\}$ and text prompt c, we aim to generate the subsequent T_{pred} frames $\{I_{T_{\text{obs}}+1}, \ldots, I_{T_{\text{obs}}+T_{\text{pred}}}\}$. This can be expressed as:

$$\hat{I}_{1:T_{\text{obs}}+T_{\text{pred}}} = \mathcal{F}_\theta \left(I_{1:T_{\text{obs}}}, c \right) \tag{5}$$

Where $T_{obs} > 1$ (*i.e.,* the number of context frames > 1), in this Video-to-Video setting, the context frames provides sufficient physical initial states (*e.g.,* position and velocity), so the subsequent motion is deterministic and can be quantitatively verified. By contrast, in $T_{obs} = 1$ setting (Image-to-Video), a single initial frame does not provide enough physical initial conditions, many future motions are plausible, making verification infeasible. We provide further analysis in the *Appendix*.

Building on this definition, we further formulate a physics-grounded reward function that quantitatively measures how well the generated videos aligns with physical laws. The details of the reward modeling are described in Sec. 3.3.

3.3 Reward Modeling

To enable physics-aware reinforcement learning, we design a new reward function guided by the principles established in the Physics-Grounded Task Definition. In reinforcement learning, the key challenge lies in designing reliable feedback for generated outputs, which directly determines the model's stability and learning direction. Many previous works [20, 57] rely on MLLM or human evaluations as the feedback signals, which are inherently subjective and coarse-grained. Such feedback cannot precisely verify the physical plausibility of generated videos and also fails to capture the relative quality among samples. Therefore, in this paper, we propose a new Physics-Grounded Metric for reward modeling, which consists of two core components: the *Trajectory Offset* and the *Collision Detection*.

Trajectory Offset. Since we establish a Physics-grounded Task Definition, where the motion of a rigid body is represented by the variation of its center coordinates, we then denote the trajectory of the object by $p_{1:T} = \{p_1, p_2, \ldots, p_T\}$, where $p_t \in \mathbb{R}^2$ indicates the center position of the object at frame t. Because rigid body motion does not involve deformation, the coordinate sequence $p_{1:T}$ can precisely reflect both positional and velocity changes over time. During data preparation, we manually annotate the object's center coordinate p_1 in the first frame. In collision scenarios, two objects are annotated, corresponding to the active and passive bodies, while in other scenarios only one object is annotated. As shown in Fig. 3, given the annotated initial coordinate p_1, both the groundtruth and generated videos are processed by SAM2 [42] to extract motion mask sequences $M^{\text{gt}}_{1:T}$ and $M^{\text{sample}}_{1:T}$. We then compute the mean pixel of each mask as the object center:

$$p^{\text{gt}}_{1:T} = \text{Center}(M^{\text{gt}}_{1:T}), p^{\text{sample}}_{1:T} = \text{Center}(M^{\text{sample}}_{1:T}) \tag{6}$$

Lastly, we quantify the Trajectory Offset(TO) between the generated and ground-truth trajectories by computing the frame-wise deviation:

$$O = \frac{1}{NT} \sum_{s=1}^{N} \sum_{t=T_{obs}}^{T} \left\| p_{t,s}^{\text{gt}} - p_{t,s}^{\text{sample}} \right\|_2 \tag{7}$$

where N denotes the number of annotated rigid bodies in the scene and $p_{t,s}^{\text{gt}}$ and $p_{t,s}^{\text{sample}}$ represent the 2D coordinates of the s-th object at frame t in the ground-truth and generated videos, respectively.

Collision Detection. Optimizing the model only with the offset in Eq. 7 causes reward hacking, where it favors simple linear motions and avoids complex collisions (Sec. 4.3). To address this, we upweight losses near collision frames, guiding the model to focus on critical physical interactions. Specifically, given the object trajectory $p_{1:T} = \{p_1, p_2, \ldots, p_T\}$, we compute the velocity sequence $v_{2:T}$, where each term is defined as $v_n = \frac{p_n - p_{n-1}}{\Delta t}$, Then, the acceleration sequence $a_{3:T}$ is derived by $a_n = \frac{v_n - v_{n-1}}{\Delta t}$. According to Newton's second law $F = ma$, the occurrence of a collision can be detected by sudden changes in acceleration. The detection method is provided in our *Appendix*. We identify the set of collision timestamps $\mathcal{C}$ and their adjacent timestamps $\mathcal{C}_{\text{adj}} = \{t \pm 1 \mid t \in \mathcal{C}\} \setminus \mathcal{C}$. To emphasize these critical moments, a temporal weight w_t is assigned to each frame t as follows:

$$w_t = \begin{cases} w^{col}, & \text{if } t \in \mathcal{C} \\ w^{adj}, & \text{if } t \in \mathcal{C}_{\text{adj}} \\ w, & \text{otherwise} \end{cases} \tag{8}$$

w^{col},$w^{adj}$and w are hyperparameters, we provide an ablation study of their effects in Sec. 4.3 and *Appendix*. Finally, the weighted trajectory offset is expressed as:

$$O_{\text{c}} = \frac{1}{NT} \sum_{s=1}^{N} \sum_{t=T_{\text{obs}}}^{T} w_t \left\| p_{t,s}^{\text{gt}} - p_{t,s}^{\text{sample}} \right\|_2, \tag{9}$$

The offset O_{c} measures trajectory discrepancies between generated and groundtruth motions, where smaller values indicate higher accuracy and stronger physical consistency. During reinforcement learning, the reward is defined as its negative, $R = -O_{\text{c}}$, so that higher rewards correspond to smaller trajectory errors, guiding the policy toward physically accurate and temporally coherent motion generation.

3.4 Mimicry-Discovery Cycle

Despite being capable of reinforcement learning, the model encounters a critical challenge (Fig. 4). When trained purely with GRPO, the model tends to perform worse on out-of-distribution samples even as it continues to improve on easier ones. This phenomenon arises because the model fails to generate high-quality

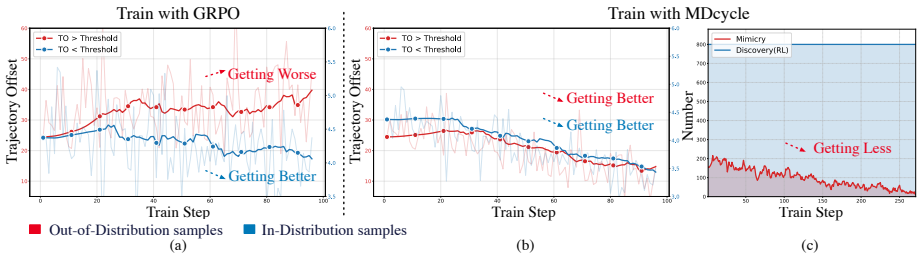

Fig. 4: (a) **GRPO Training loss** for samples of varying quality. (b) **MDcycle Training loss** for samples of varying quality. (c) **Number of samples** assigned to the Mimicry and Discovery branches throughout `MDcycle`.

outputs in these challenging scenarios, leaving it insufficiently prepared for effective exploration in reinforcement learning. To address this issue, we apply the Flow Matching loss to these out-of-distribution cases, providing fine-grained pixel-level supervision that improves convergence. As evidenced in Fig. 4(b,c), the model converges on samples from both distributions, and the proportion of *"Mimicry"* branch samples gradually decreases during training, indicating that the model progressively acquires the capability for *"Discovery"*.

Specifically, given a textual condition c and a context video $I_{1:T_{\text{obs}}}$, the model generates a group of video samples $\{x_o^i\}_{i=1}^G$. Each sample is evaluated by the evaluate function O_c described above, and we compute per-sample advantage value $\{A^i\}_{i=1}^G$ using Eq. (3). Additionally, we compute the group-average trajectory offset $\bar{O}_c = \frac{1}{G}\sum_{i=1}^G O_c(x_0^i, c)$ to assess the performance on this group. We introduce a hyperparameter Threshold to control the strategy switch: when $\bar{O}_c >$ Threshold, it indicates that the model performs poorly on this case, and we add an additional Flow Matching loss term L_M to guide optimization. Otherwise, the training relies primarily on the RL objective $L_D = -J(\theta)$ Eq. (4). Formally, the unified optimization objective is defined as:

$$L = L_D + \alpha L_M, \tag{10}$$

$$\text{where} \quad \alpha = \begin{cases} 1, & \text{if } \bar{O}_c > \text{Threshold} \\ 0, & \text{otherwise.} \end{cases}$$

`MDcycle` essentially operates under a reinforcement learning framework, which is more complex than standard fully finetune or PEFT such as LoRA. Due to space limitations, we provide only a brief description here and include the full *pseudocode* in the *Appendix*.

3.5 Training Pipeline

We follow a **I2V-V2V-RL** training pipeline that gradually transitions the model from conventional image-to-video generation to physics-aware video-to-video generation. We start from a pretrained diffusion transformer with strong visual priors and adapt it to V2V in the first stage. Specifically, we replace the

Table 1: Quantitative comparisons with existing methods on Vbench, VideoPhy-2 (SA, PC) and `PhysRVGBench` (IoU, TO). The best results are shown in **bold**, and the second best are underlined.

Model	Subj. Cons.↑	Back. Cons.↑	Image Qual.↑	Moti. Smoo.↑	Dyna. Degr.↑	Aest. Qual.↑	Temp. Flic.↑	Total Score↑	SA↑	PC↑	IoU↑	TO↓
Wan2.2 14B	94.75	95.94	60.66	99.21	50.00	38.61	98.64	76.83	0.64	0.34	0.12	162.40
Kling2.5	95.32	95.56	64.85	99.47	**57.14**	40.29	98.76	78.77	0.70	0.41	0.23	103.22
HunyuanVideo	95.99	96.75	59.80	99.47	46.00	41.28	99.06	76.97	0.60	0.32	0.10	181.62
Magi-1	**97.03**	97.40	60.66	99.57	48.28	37.67	99.55	77.16	0.67	0.38	0.27	113.42
VideoMAR	92.86	92.43	57.92	97.59	49.78	35.08	94.44	74.31	0.62	0.32	0.31	109.43
CosmosPredict2	93.88	97.35	60.27	98.71	52.74	37.56	97.88	76.91	0.70	0.39	0.33	79.67
`PhysRVG`	96.97	**97.74**	**64.96**	**99.63**	52.00	**41.36**	**99.58**	**78.89**	**0.76**	**0.44**	**0.64**	**15.03**

original image condition with the first $T_{\text{obs}} = 5$ frames of the input video and perform full-parameter fine-tuning on our training set using the Flow Matching loss, enabling basic temporal generation capability. In the second stage, we focus on improving physical consistency through training under the `MDcycle` framework, using a parameter-efficient weight setup (LoRA) and reinforcement feedback guided by the Physics-Grounded Metric. This staged pipeline ensures stable optimization while introducing physics-aware objectives.

4 Experiment

4.1 Experimental Settings

Dataset. Our training data consist of both open-source and proprietary video collections, totaling approximately 10M samples. The open-source portion includes Panda-70M, InternVid, and WebVid-10M, providing diverse motion scenes and visual contexts. To supplement these, we additionally collect proprietary video data, including real-world competition footage, synthetic videos recorded in video games, and real experiments captured using our own recording equipment. For high-quality rigid-body motion data, which are relatively scarce, we manually curate and annotate about 700 video samples covering various motion types such as collision, rolling, pendulum, and free fall. Among them, around 50 videos are separated as the benchmark test set, which is completely excluded from training. Representative samples are provided in the *Appendix*.

Evaluation Metrics. We adopt VBench [19] to evaluate visual fidelity and VideoPhy-2 [4] to evaluate physical realism in generated videos. To further evaluate physical realism in the rigid body motion, we employ `PhysRVGBench`, which measures the discrepancy between generated and ground-truth motions using Intersection-over-Union (IoU) and Trajectory Offset (TO) to capture spatial overlap and trajectory deviation. Details of our newly proposed IoU and TO are provided in *Appendix*.

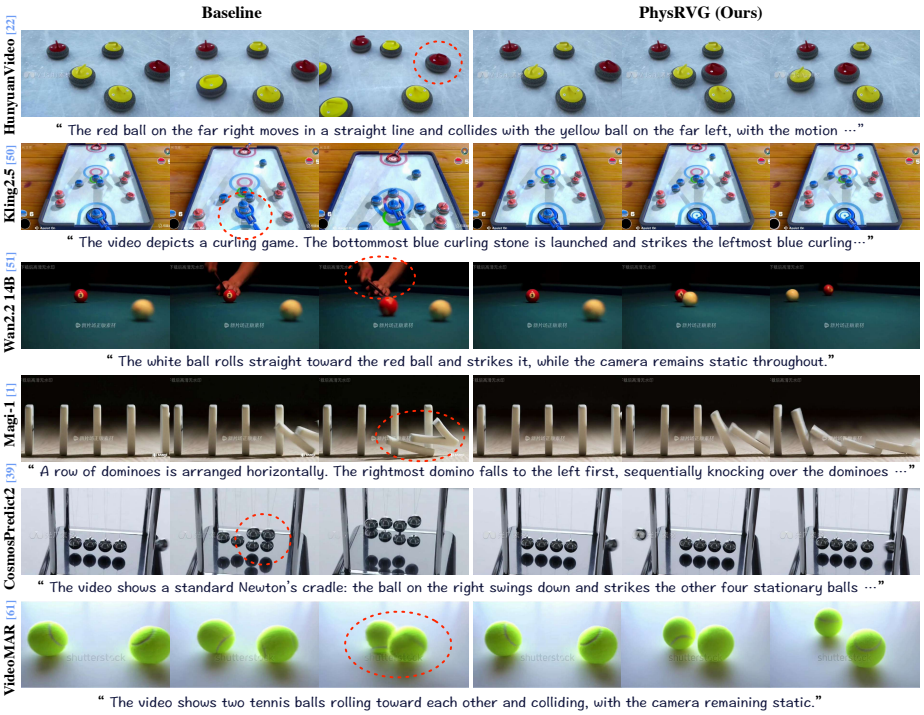

Fig. 5: Qualitative comparisons with existing methods. We include the original videos for all cases in the *Supplementary Materials*.

4.2 Experimental Results

General Comparisons. We compare our approach against current state-of-the-art I2V and V2V models using 50 video samples from the test set. To ensure a fair comparison, we standardize the video generation settings and carefully design prompts to guide models in generating the corresponding motion dynamics. Tab. 1 presents quantitative results on VBench [19], VideoPhy-2 [4], and PhysRVGBench. Our method surpasses SOTA models across all benchmark, with particularly significant gains on VideoPhy-2 and PhysRVGBench. This indicates that while current SOTA models excel in visual fidelity, they struggle significantly with physical realism. We provide visual comparisons in Fig. 5, demonstrating that our method generates videos with greater physical plausibility. Rows 1-2 illustrate composite motions involving linear movement and collisions. We observe that HunyuanVideo [22] produces incorrect collision trajectories for the curling stone, and Kling2.5 [50] fails to accurately comprehend the text prompt. Rows 3 show that an unexpected human appears in the results generated by Wan2.2 [51], and the billiard motion is chaotic. These errors are consistent with a human-centric dataset bias that favors generating people over enforcing physical plausibility. Rows 4-6 depict more complex rigid body collisions. In the outputs of Magi-1 [1], CosmosPredict2 [39], and VideoMAR [61], objects become entangled or exhibit irregular motion. In contrast, `PhysRVG` pre-

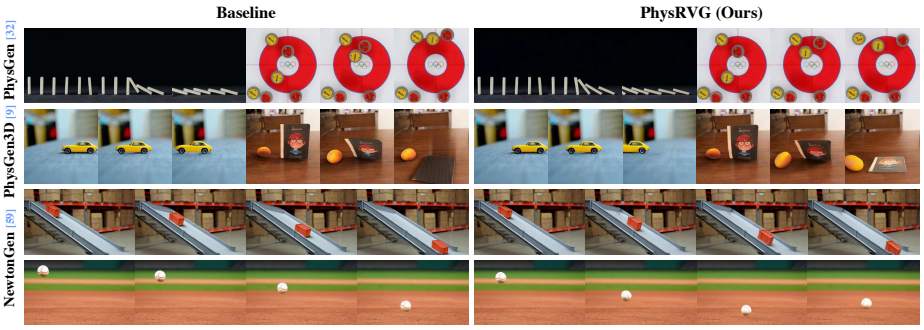

Fig. 6: Comparison with Physics-Aware Methods.

serves physical integrity and motion coherence across these settings, accurately capturing rolling and collision behaviors and reflecting a sound grasp of rigid body physics. We include the original videos in the *Supplementary Materials*, all results are produced with a single random seed without manual selection.

Comparison with Physics-Aware Methods. We compare our method with several physics-aware approaches in Fig. 6, including PhysGen [32], PhysGen3D [9], and NewtonGen [62]. However, due to substantial differences in problem settings, a strictly fair comparison is difficult. Specifically, our method extrapolates future frames conditioned on a few context frames, whereas these methods require simulating the entire motion trajectory in advance to obtain a motion prior. Here, we mainly investigate whether our method can generalize to their scenarios. To this end, we collect videos generated by these methods, use their first five frames as context frames, and generate the subsequent frames. We find `PhysRVG` can infer plausible motion dynamics from limited initial physical conditions, and remains effective across different resolutions and out-of-distribution scenarios.

4.3 Ablation Study

Analysis of Key Design Components. We conduct an overall quantitative evaluation of key design components in Tab. 2 and revisit the role of them. The finetuning stage converts the I2V model into a V2V model, enabling it to generate coherent frames conditioned on the given context frames. Finetuning alone causes the model to produce motion with random trajectories, so we use reinforcement learning to steer the model toward physically plausible trajectories. However, applying GRPO alone makes training unstable and often yields suboptimal results. MDcycle effectively addresses this issue and we will analyze it further later. Visual inspection shows that the model still struggles at very subtle moments of collision. We therefore extend the reward with Collision Detection to better optimize these critical instants and we will discuss it in the following. As shown in Tab. 2, this series of strategies leads to consistent improvements.

MDcycle. We further conduct an ablation study to evaluate `MDcycle`. As shown in Fig. 7 and Fig. 8, `MDcycle` achieves smoother convergence and higher final performance than GRPO. The reward curves in Fig. 7(a) show that `MDcycle`

Table 2: Ablation study of the key design components on Vbench, VideoPhy-2 (SA, PC) and `PhysRVGBench` (IoU, TO). From top to bottom, modules are added sequentially to the previous configuration. (*e.g.*, "+GRPO" = Baseline + FT + GRPO).

Model	Subj. Cons.↑	Back. Cons.↑	Image Qual.↑	Moti. Smoo.↑	Dyna. Degr.↑	Aest. Qual.↑	Temp. Flic.↑	Total Score↑	SA↑	PC↑	IoU↑	TO↓
Baseline	86.64	90.35	59.06	97.87	56.00	37.02	97.11	74.86	0.57	0.21	0.15	162.78
+Fine-tuning(FT)	94.77	97.48	61.23	99.08	51.02	39.77	98.82	77.45	0.63	0.34	0.41	48.60
+GRPO	95.21	97.36	62.27	99.32	51.82	40.23	99.18	77.91	0.69	0.38	0.52	26.38
+MDcycle	96.17	97.49	63.88	99.75	51.79	41.19	99.44	78.53	0.74	0.43	0.62	19.47
+Collision Detect	96.97	97.74	64.96	99.63	52.00	41.36	99.58	78.89	0.76	0.44	0.64	15.03

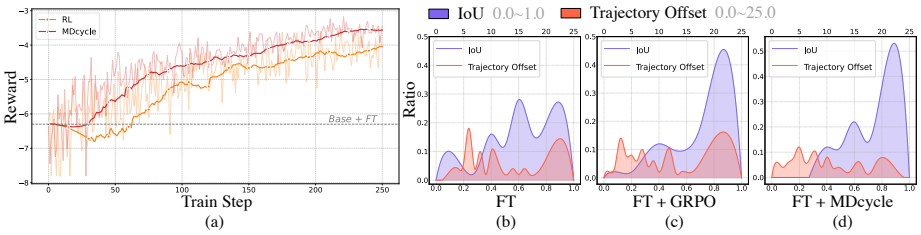

Fig. 7: (a) **Reward Curve** of GRPO and `MDcycle`. (b)(c)(d) **Metric Distribution** under different training strategies.

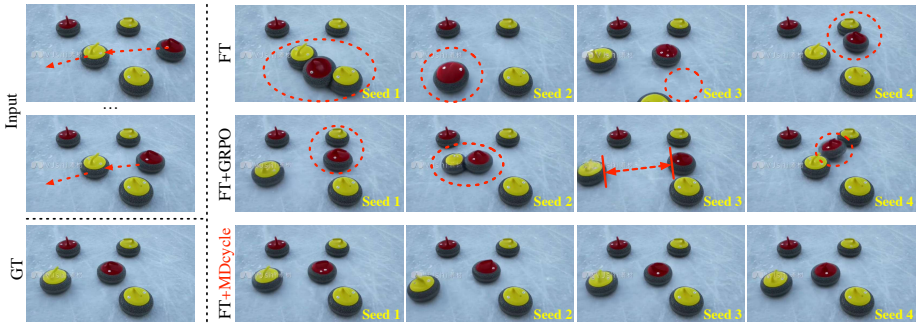

Fig. 8: Ablation Study of the MDcycle.

converges steadily and reaches higher rewards, while pure GRPO training remains unstable during the initial 50 steps. This instability arises because the model cannot produce reliable outputs in out-of-distribution scenarios within limited exploratory attempts. A similar pattern is observed in the metric distributions of sampled results, as shown in Fig. 7(b,c,d). Compared with FT and FT+GRPO, `MDcycle` increases the proportion of high-quality samples, such as those with TO in [0, 5] and IoU in [0.8, 1.0], and significantly reduces low-quality cases, such as those with TO in [15, 25] and IoU in [0.0, 0.4]. Together, these results demonstrate that `MDcycle` provides greater training stability and stronger physical consistency. Fig. 8 shows results in a complex scenario involving large range motion and collisions. While FT and FT+GRPO produce random trajectories under different random seeds in this setting, indicating that the model

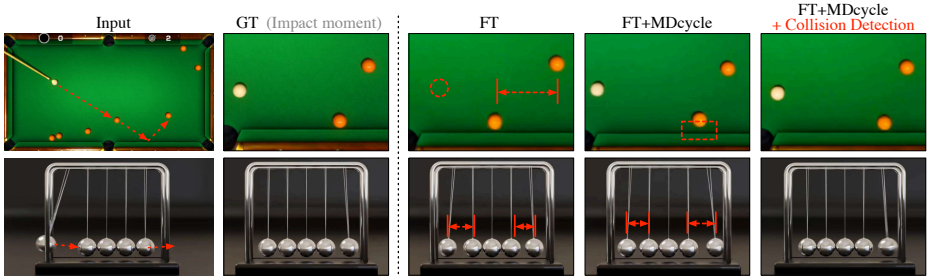

Fig. 9: Ablation Study of the Collision Detection.

Table 3: Hyperparameter Analysis. All comparisons are conducted under the same data and GPU-hours budget. (a) SDE Interval. (b) Noise Intensity $\boldsymbol{\sigma}_t$. (c) Threshold(TH) in Eq. (10). The best configuration is marked in gray.

SDE Window	NFE	IoU↑	TO↓
75 % – 100 %	2	**0.64**	**15.03**
50 % – 100 %	2	0.63	15.58
25 % – 100 %	2	0.61	17.07
0 % – 100 %	16	0.55	27.24

(a)

σ_t	IoU↑	TO↓
0.2	0.61	17.31
0.6	0.64	15.83
1.0	**0.64**	**15.03**
1.4	0.62	16.91

(b)

TH	IoU↑	TO↓
4	0.57	22.39
8	**0.64**	**15.03**
12	0.61	16.97

(c)

has not truly learned the physical rules, FT+MDcycle generates consistent and physically plausible trajectories across seeds. This suggests that MDcycle steers generation toward physically plausible trajectories.

Collision Detection. We conduct an ablation study to evaluate the effect of collision-aware design in the reward model. Fig. 9 presents the qualitative results, where the FT model produces artifacts such as object disappearance and inaccurate trajectories, and MDcycle improves motion smoothness but fails to handle collisions correctly. Incorporating collision-aware rewards eliminates these issues and yields physically consistent motions.

SDE Interval. We conduct a hyperparameter study on the hybrid SDE-ODE sampling strategy to evaluate its effect on RL training efficiency. Within each sampling window, two consecutive steps use an SDE sampler, and all remaining sampling steps use an ODE sampler. Results in Tab. 3a show that SDE sampling performs best in high-noise regions, indicating that stochastic exploration at early noisy stages enhances semantic learning and overall stability.

Noise Intensity. σ_t in Eq. (2) controls the noise intensity, which determines how broadly the model explores the solution space. In most cases, the value is set lower than the optimal 1.0 reported in Tab. 3b (*e.g.*, 0.3 in DanceGRPO), since excessive noise may degrade the model's original performance and training stability. In `PhysRVG`, we encourage active exploration of physical knowledge, making a higher noise ratio desirable. Stable training under this setting is achieved because the V2V process provides an unbiased initialization, ensuring reliable convergence.

Threshold. We analyze the Threshold in Eq. (10), which regulates the balance between mimicry and discovery. As shown in Tab. 3c, a small threshold causes over-mimicry and limits exploration, whereas a large one weakens guidance and destabilizes training. A proper threshold allows the model to adaptively transition from mimicry-driven stabilization to discovery-driven exploration, achieving a balanced and self-regulated learning process.

5 Conclusion

We present `PhysRVG`, a unified reinforcement learning framework with physically verifiable rewards. It enhances the physical modeling ability of video generative models. Conventional pretrain–finetune paradigms focus on pixel reconstruction and perceptual quality while neglecting physical realism. To overcome this, we introduce a physics-grounded metric that measures motion fidelity and integrate abstract physical knowledge through `MDcycle`. We also develop `PhysRVGBench`, a benchmark for evaluating rigid body motion quality. Extensive experiments demonstrate the effectiveness of our approach. Future works, limitations, and ethical considerations are provided in *Appendix*.

Acknowledgements

This work was supported by Ant Group Research Intern Program. This research was also supported by Zhejiang Provincial Natural Science Foundation of China under Grant No. LD24F020007.

References

1. ai, S., Teng, H., Jia, H., Sun, L., Li, L., Li, M., Tang, M., Han, S., Zhang, T., Zhang, W.Q., Luo, W., Kang, X., Sun, Y., Cao, Y., Huang, Y., Lin, Y., Fang, Y., Tao, Z., Zhang, Z., Wang, Z., Liu, Z., Shi, D., Su, G., Sun, H., Pan, H., Wang, J., Sheng, J., Cui, M., Hu, M., Yan, M., Yin, S., Zhang, S., Liu, T., Yin, X., Yang, X., Song, X., Hu, X., Zhang, Y., Li, Y.: Magi-1: Autoregressive video generation at scale (2025)
2. Aira, L.S., Montanaro, A., Aiello, E., Valsesia, D., Magli, E.: Motioncraft: Physics-based zero-shot video generation (2024), https://arxiv.org/abs/2405.13557
3. Banerjee, C., Nguyen, K., Fookes, C., Karniadakis, G.: Physics-informed computer vision: A review and perspectives (2024), https://arxiv.org/abs/2305.18035
4. Bansal, H., Peng, C., Bitton, Y., Goldenberg, R., Grover, A., Chang, K.W.: Videophy-2: A challenging action-centric physical commonsense evaluation in video generation (2025), https://arxiv.org/abs/2503.06800
5. Black, K., Janner, M., Du, Y., Kostrikov, I., Levine, S.: Training diffusion models with reinforcement learning. arXiv preprint arXiv:2305.13301 (2023)
6. Bordes, F., Garrido, Q., Kao, J.T., Williams, A., Rabbat, M., Dupoux, E.: Intphys 2: Benchmarking intuitive physics understanding in complex synthetic environments (2025), https://arxiv.org/abs/2506.09849
7. Cai, Y., Li, K., Jia, M., Wang, J., Sun, J., Liang, F., Chen, W., Juefei-Xu, F., Wang, C., Thabet, A., Dai, X., Ju, X., Yuille, A., Hou, J.: Phygdpo: Physics-aware groupwise direct preference optimization for physically consistent text-to-video generation (2026), https://arxiv.org/abs/2512.24551
8. Cao, J., Guan, S., Ge, Y., Li, W., Yang, X., Ma, C.: Neural material adaptor for visual grounding of intrinsic dynamics (2024), https://arxiv.org/abs/2410.08257
9. Chen, B., Jiang, H., Liu, S., Gupta, S., Li, Y., Zhao, H., Wang, S.: Physgen3d: Crafting a miniature interactive world from a single image (2025), https://arxiv.org/abs/2503.20746
10. Chen, G., Lin, D., Yang, J., Lin, C., Zhu, J., Fan, M., Zhang, H., Chen, S., Chen, Z., Ma, C., Xiong, W., Wang, W., Pang, N., Kang, K., Xu, Z., Jin, Y., Liang, Y., Song, Y., Zhao, P., Xu, B., Qiu, D., Li, D., Fei, Z., Li, Y., Zhou, Y.: Skyreels-v2: Infinite-length film generative model (2025), https://arxiv.org/abs/2504.13074
11. Chen, J., Zhao, Y., Yu, J., Chu, R., Chen, J., Yang, S., Wang, X., Pan, Y., Zhou, D., Ling, H., Liu, H., Yi, H., Zhang, H., Li, M., Chen, Y., Cai, H., Fidler, S., Luo, P., Han, S., Xie, E.: Sana-video: Efficient video generation with block linear diffusion transformer (2025), https://arxiv.org/abs/2509.24695
12. DeepSeek-AI: Deepseek-v3 technical report (2024), https://arxiv.org/abs/2412.19437
13. DeepSeek-AI: Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning (2025), https://arxiv.org/abs/2501.12948
14. Gao, L., Schulman, J., Hilton, J.: Scaling laws for reward model overoptimization. In: International Conference on Machine Learning. pp. 10835–10866. PMLR (2023)
15. Gong, B., Huang, S., Feng, Y., Zhang, S., Li, Y., Liu, Y.: Check locate rectify: A training-free layout calibration system for text-to-image generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6624–6634 (2024)
16. Gong, B., Tan, S., Feng, Y., Xie, X., Li, Y., Chen, C., Zheng, K., Shen, Y., Zhao, D.: Uknow: A unified knowledge protocol with multimodal knowledge graph datasets

for reasoning and vision-language pre-training. Advances in Neural Information Processing Systems **37**, 9612–9633 (2024)
17. He, X., Fu, S., Zhao, Y., Li, W., Yang, J., Yin, D., Rao, F., Zhang, B.: Tempflow-grpo: When timing matters for grpo in flow models. arXiv preprint arXiv:2508.04324 (2025)
18. Hu, Y., Wang, L., Liu, X., Chen, L.H., Guo, Y., Shi, Y., Liu, C., Rao, A., Wang, Z., Xiong, H.: Simulating the real world: A unified survey of multimodal generative models (2025), https://arxiv.org/abs/2503.04641
19. Huang, Z., He, Y., Yu, J., Zhang, F., Si, C., Jiang, Y., Zhang, Y., Wu, T., Jin, Q., Chanpaisit, N., et al.: Vbench: Comprehensive benchmark suite for video generative models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 21807–21818 (2024)
20. Ji, S., Chen, X., Tao, X., Wan, P., Zhao, H.: Physmaster: Mastering physical representation for video generation via reinforcement learning (2025), https://arxiv.org/abs/2510.13809
21. Jiang, C., Schroeder, C., Teran, J., Stomakhin, A., Selle, A.: The material point method for simulating continuum materials. In: ACM SIGGRAPH 2016 Courses. SIGGRAPH '16, Association for Computing Machinery, New York, NY, USA (2016). https://doi.org/10.1145/2897826.2927348, https://doi.org/10.1145/2897826.2927348
22. Kong, W., Tian, Q., Zhang, Z., Min, R., Dai, Z., Zhou, J., Xiong, J., Li, X., Wu, B., Zhang, J., et al.: Hunyuanvideo: A systematic framework for large video generative models. arXiv preprint arXiv:2412.03603 (2024)
23. Li, C., Michel, O., Pan, X., Liu, S., Roberts, M., Xie, S.: Pisa experiments: Exploring physics post-training for video diffusion models by watching stuff drop. In: ICML (2025)
24. Li, J., Cui, Y., Huang, T., Ma, Y., Fan, C., Yang, M., Zhong, Z.: Mixgrpo: Unlocking flow-based grpo efficiency with mixed ode-sde. arXiv preprint arXiv:2507.21802 (2025)
25. Li, Z., Tucker, R., Snavely, N., Holynski, A.: Generative image dynamics (2024), https://arxiv.org/abs/2309.07906
26. Li, Z., Yu, H.X., Liu, W., Yang, Y., Herrmann, C., Wetzstein, G., Wu, J.: Wonderplay: Dynamic 3d scene generation from a single image and actions (2025), https://arxiv.org/abs/2505.18151
27. Lin, M., Wang, X., Wang, Y., Wang, S., Dai, F., Ding, P., Wang, C., Zuo, Z., Sang, N., Huang, S., Wang, D.: Exploring the evolution of physics cognition in video generation: A survey (2025), https://arxiv.org/abs/2503.21765
28. Lin, Y., Lin, C., Xu, J., Mu, Y.: Omniphysgs: 3d constitutive gaussians for general physics-based dynamics generation (2025), https://arxiv.org/abs/2501.18982
29. Lipman, Y., Chen, R.T.Q., Ben-Hamu, H., Nickel, M., Le, M.: Flow matching for generative modeling (2023), https://arxiv.org/abs/2210.02747
30. Liu, D., Zhang, J., Dinh, A.D., Park, E., Zhang, S., Mian, A., Shah, M., Xu, C.: Generative physical ai in vision: A survey (2025), https://arxiv.org/abs/2501.10928
31. Liu, J., Liu, G., Liang, J., Li, Y., Liu, J., Wang, X., Wan, P., Zhang, D., Ouyang, W.: Flow-grpo: Training flow matching models via online rl. arXiv preprint arXiv:2505.05470 (2025)
32. Liu, S., Ren, Z., Gupta, S., Wang, S.: Physgen: Rigid-body physics-grounded image-to-video generation. In: European Conference on Computer Vision (ECCV) (2024)

33. Liu, X., Gong, C., Liu, Q.: Flow straight and fast: Learning to generate and transfer data with rectified flow. In: ICML (2022)
34. Lv, J., Huang, Y., Yan, M., Huang, J., Liu, J., Liu, Y., Wen, Y., Chen, X., Chen, S.: Gpt4motion: Scripting physical motions in text-to-video generation via blender-oriented gpt planning (2024), https://arxiv.org/abs/2311.12631
35. Ma, G., Huang, H., Yan, K., Chen, L.: Step-video-t2v technical report: The practice, challenges, and future of video foundation model (2025), https://arxiv.org/abs/2502.10248
36. Macklin, M., Müller, M., Chentanez, N.: Xpbd: position-based simulation of compliant constrained dynamics. In: Proceedings of the 9th International Conference on Motion in Games. p. 49–54. MIG '16, Association for Computing Machinery, New York, NY, USA (2016). https://doi.org/10.1145/2994258.2994272, https://doi.org/10.1145/2994258.2994272
37. Meng, S., Luo, Y., Liu, P.: Grounding creativity in physics: A brief survey of physical priors in aigc (2025), https://arxiv.org/abs/2502.07007
38. Müller, M., Heidelberger, B., Hennix, M., Ratcliff, J.: Position based dynamics. J. Vis. Comun. Image Represent. **18**(2), 109–118 (Apr 2007). https://doi.org/10.1016/j.jvcir.2007.01.005, https://doi.org/10.1016/j.jvcir.2007.01.005
39. NVIDIA, :, Agarwal, N., Ali, A.: Cosmos world foundation model platform for physical ai (2025)
40. Ouyang, L., Wu, J., Jiang, X.: Training language models to follow instructions with human feedback (2022), https://arxiv.org/abs/2203.02155
41. Rafailov, R., Sharma, A., Mitchell, E., Manning, C.D., Ermon, S., Finn, C.: Direct preference optimization: Your language model is secretly a reward model (2023), https://arxiv.org/abs/2305.18290
42. Ravi, N., Gabeur, V., Hu, Y.T., Hu, R., Ryali, C., Ma, T., Khedr, H., Rädle, R., Rolland, C., Gustafson, L., Mintun, E., Pan, J., Alwala, K.V., Carion, N., Wu, C.Y., Girshick, R., Dollár, P., Feichtenhofer, C.: Sam 2: Segment anything in images and videos. arXiv preprint arXiv:2408.00714 (2024), https://arxiv.org/abs/2408.00714
43. Rezatofighi, H., Tsoi, N., Gwak, J., Sadeghian, A., Reid, I., Savarese, S.: Generalized intersection over union: A metric and a loss for bounding box regression (2019), https://arxiv.org/abs/1902.09630
44. Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O.: Proximal policy optimization algorithms (2017), https://arxiv.org/abs/1707.06347
45. Shao, Z., Wang, P., Zhu, Q., Xu, R., Song, J., Bi, X., Zhang, H., Zhang, M., Li, Y.K., Wu, Y., Guo, D.: Deepseekmath: Pushing the limits of mathematical reasoning in open language models (2024), https://arxiv.org/abs/2402.03300
46. Shi, S., Gong, B., Chen, X., Zheng, D., Tan, S., Yang, Z., Li, Y., He, J., Zheng, K., Chen, J., et al.: Motionstone: Decoupled motion intensity modulation with diffusion transformer for image-to-video generation. CVPR (2025)
47. Tan, S., Gong, B., Feng, Y., Zheng, K., Zheng, D., Shi, S., Shen, Y., Chen, J., Yang, M.: Mimir: Improving video diffusion models for precise text understanding. CVPR (2025)
48. Tan, X., Jiang, Y., Li, X., Zong, Z., Xie, T., Yang, Y., Jiang, C.: Physmotion: Physics-grounded dynamics from a single image (2024), https://arxiv.org/abs/2411.17189
49. Team, G.: Mochi 1. https://github.com/genmoai/models (2024)
50. Team, K., Chen, J., Ci, Y., Du, X., Feng, Z., Gai, K., Guo, S., Han, F., He, J., He, K., Hu, X., Hu, X., Jiang, B., Kong, F., Li, H., Li, J., Li, Q., Li, S., Li, X., Li,

Y., Liang, J., Liao, B., Liao, Y., Lin, W., Liu, Q., Liu, X., Liu, Y., Liu, Y., Lu, S., Mao, H., Mao, Y., Ouyang, H., Qin, W., Shi, W., Shi, X., Su, L., Sun, H., Sun, P., Wan, P., Wang, C., Wang, C., Wang, M., Wang, Q., Wang, R., Wang, X., Wang, X., Wang, Z., Wei, M., Wen, T., Wu, G., Wu, X., Wu, Z., Xie, D., Xiong, Y., Xu, Y., Yang, S., Yang, Z., Ye, W., Yuan, Z., Zhang, S., Zhang, S., Zhang, Y., Zhang, Y., Zhao, W., Zhou, R., Zhou, Y., Zhu, G., Zhu, Y.: Kling-omni technical report (2025), https://arxiv.org/abs/2512.16776

51. Wan, T., Wang, A., Ai, B.: Wan: Open and advanced large-scale video generative models (2025), https://arxiv.org/abs/2503.20314
52. Wang, J., Ma, A., Cao, K., Zheng, J., Zhang, Z., Feng, J., Liu, S., Ma, Y., Cheng, B., Leng, D., Yin, Y., Liang, X.: Wisa: World simulator assistant for physics-aware text-to-video generation (2025), https://arxiv.org/abs/2502.08153
53. Wei, Y., Zhang, S., Yuan, H., Gong, B., Tang, L., Wang, X., Qiu, H., Li, H., Tan, S., Zhang, Y., et al.: Dreamrelation: Relation-centric video customization. ICCV (2025)
54. Xiang, K., Zhang, T.J., Huang, Y., He, J., Liu, Z., Tang, Y., Zhou, R., Luo, L., Wen, Y., Chen, X., Lin, B., Han, J., Xu, H., Li, H., Dong, B., Liang, X.: Aligning perception, reasoning, modeling and interaction: A survey on physical ai (2025), https://arxiv.org/abs/2510.04978
55. Xie, T., Zhao, Y., Jiang, Y., Jiang, C.: Physanimator: Physics-guided generative cartoon animation. In: CVPR (2025)
56. Xie, T., Zong, Z., Qiu, Y., Li, X., Feng, Y., Yang, Y., Jiang, C.: Physgaussian: Physics-integrated 3d gaussians for generative dynamics. In: CVPR (2023)
57. Xue, Q., Yin, X., Yang, B., Gao, W.: Phyt2v: Llm-guided iterative self-refinement for physics-grounded text-to-video generation (2025), https://arxiv.org/abs/2412.00596
58. Xue, Z., Wu, J., Gao, Y., Kong, F., Zhu, L., Chen, M., Liu, Z., Liu, W., Guo, Q., Huang, W., et al.: Dancegrpo: Unleashing grpo on visual generation. arXiv preprint arXiv:2505.07818 (2025)
59. Yang, A., Yang, B., Hui, B., Zheng, B.: Qwen2 technical report (2024), https://arxiv.org/abs/2407.10671
60. Yang, Z., Teng, J., Zheng, W., Ding, M., Huang, S., Xu, J., Yang, Y., Hong, W., Zhang, X., Feng, G., Yin, D., Zhang, Y., Wang, W., Cheng, Y., Xu, B., Gu, X., Dong, Y., Tang, J.: Cogvideox: Text-to-video diffusion models with an expert transformer (2025), https://arxiv.org/abs/2408.06072
61. Yu, H., Gong, B., Yuan, H., Zheng, D., Chai, W., Chen, J., Zheng, K., Zhao, F.: Videomar: Autoregressive video generatio with continuous tokens (2025), https://arxiv.org/abs/2506.14168
62. Yuan, Y., Wang, X., Wickremasinghe, T., Nadir, Z., Ma, B., Chan, S.: Newtongen: Physics-consistent and controllable text-to-video generation via neural newtonian dynamics. arXiv preprint arXiv: 2509.21309 (2025)
63. Zhang, T., Yu, H.X., Wu, R., Feng, B.Y., Zheng, C., Snavely, N., Wu, J., Freeman, W.T.: PhysDreamer: Physics-based interaction with 3d objects via video generation. In: European Conference on Computer Vision. Springer (2024)
64. Zhou, Y., Ling, P., Bu, J., Wang, Y., Zang, Y., Wang, J., Niu, L., Zhai, G.: G2rpo: Granular grpo for precise reward in flow models. arXiv preprint arXiv:2510.01982 (2025)