Efficient Off-Policy RL for Video Generation via Forward-Consistent Reward Matching

Anonymous Author(s)
Affiliation
Address
email

Abstract

Reinforcement learning (RL) post-training aligns diffusion-based generators with human preferences, yet existing RL methods suffer from poor compatibility with off-policy learning and few-step distilled models. These limitations are especially severe in the video generation area, as practical video generation pipelines often rely on few-step distilled generators. Furthermore, due to complex spatial-temporal dynamics and higher dimensions, near-on-policy video rollouts are both expensive to collect and often imperfect. Relying on such rollouts alone can amplify artifacts and is prone to reward hacking. To address these issues, we propose `Forward-Consistent Reward Matching` (`FCRM`), an efficient off-policy RL framework for video generation. `FCRM` converts the forward denoising loss into a positive loss-induced score and formulates the reward alignment as a one-step GFlowNet matching problem. The resulting residual is pointwise in a clean sample space that naturally supports off-policy learning and few-step generators. To avoid biased gradients, we introduce a double-sampling estimator for the squared residual objective. Theoretically, minimizing the proposed matching residual bounds the KL divergence between the learned distribution and the optimal reward-tilted distribution. Experiments on standard video generation benchmarks validate `FCRM` across online, replay, offline, and few-step settings and outperform SOTA methods.

1 Introduction

Video generation models have seen rapid advancements, largely driven by diffusion models and flow matching techniques [16, 24, 35] that iteratively map simple noise distributions to complex, high-dimensional videos. These advances have enabled increasingly capable video generators [5, 7, 8, 38]. However, these models use likelihood- or matching-based pretraining objectives, which do not directly optimize for human preference, prompt faithfulness, or aesthetic quality [4, 13]. Reinforcement learning (RL) post-training has emerged as a crucial step to bridge this gap, enabling models to optimize for external reward signals.

Early diffusion RL methods formulate the reverse sampling process as a multi-step Markov Decision Process, yielding policy-gradient-style methods such as DDPO, DPOK, and, in the flow-matching setting, Flow-GRPO [4, 13, 25]. However, such a reverse-process RL relies on explicit estimation of the backward trajectory's likelihood, which requires costly multi-step SDE rollouts during training. More recent work instead formulates post-training in clean-sample or forward-process space, e.g. Advantage Weighted Matching (AWM) and DiffusionNFT [42, 48]. This paradigm maintains what we call *forward consistency*. The RL objective is defined via the forward noising process used in the pretraining, rather than through a discretized reverse-time trajectory. This makes the objective less tied to the particular reverse-time solver used at deployment.

Accepted at ICML 2026 Workshop on Decision-Making from Offline Datasets to Online Adaptation: Black-Box Optimization to Reinforcement Learning.

However, existing diffusion RL post-training methods have been designed and experimentally validated primarily in the image domain. Video generation presents unique challenges that require a different paradigm. From an inference perspective, video generation is substantially more computationally expensive than image generation. Consequently, practical video generation pipelines often rely on few-step generators obtained via distillation [12, 32, 45, 47]. Some recent work proposes jointly optimizing distillation and RL [20]. However, we argue that models require continuous improvement post-deployment. RL should serve as a flexible component applied after step distillation, necessitating RL methods that are *compatible with few-step models*. From a training perspective, video models are more susceptible to visual artifacts and reward hacking than image models due to complex spatial-temporal dynamics. Video evaluation encompasses numerous subjective and complex dimensions, such as motion naturalness, temporal consistency, and aesthetic quality. A single proxy reward model struggles to capture perfectly. Online rollouts can easily exploit the blind spots of these imperfect reward models, causing artifacts to be gradually reinforced during training. Therefore, video RL training requires a natural mechanism to *incorporate high-quality off-policy (or offline) data*, which can anchor the learned distribution and effectively mitigate reward hacking.

Unfortunately, existing diffusion RL methods cannot directly satisfy these requirements for video generation. Reverse-process RL methods require explicit trajectory likelihoods, which makes them incompatible with few-step distilled generators. While recent forward-process methods bypass the need for multi-step rollouts, they still struggle with off-policy learning. In particular, AWM plugs the forward surrogate into a GRPO-style objective, which is still near-on-policy. DiffusionNFT interprets its supervised surrogate as an implicit policy-improvement operator, but this equivalence relies on solving the surrogate to optimality under a fixed behavior policy. In practice, the behavior policy is typically an Exponential Moving Average (EMA) sampler that changes throughout training, making the target non-stationary. More generally, neither perspective formulates RL post-training as pointwise matching with a specified target distribution in a manner that is inherently compatible with off-policy learning. Consequently, these approaches cannot fully exploit offline video datasets or temporally outdated replay samples within the RL training pipeline.

To overcome these challenges, we propose `FCRM` (`Forward-Consistent Reward Matching`), a forward-consistent approach that operates purely in a forward loss-induced sample space. We first convert the per-sample forward denoising loss into a positive score on clean samples, which defines an unnormalized terminal density. Then we formulate RL post-training as a *one-step GFlowNet matching problem* and match the normalized density to a reward-tilted target via the standard GFlowNet detailed balance condition [3]. This gives a conceptually simple objective with three practical properties relevant for video generation. It is *forward consistent* by construction. Because the target is defined pointwise in a clean-sample space, it supports *off-policy learning by design*, allowing it to seamlessly adopt replay or offline data. It is *fully compatible with few-step distilled generators*. Furthermore, although our method is designed in the forward loss-induced space, it is not merely a heuristic surrogate disconnected from the original clean-sample RL objective. We theoretically derive a bound demonstrating that minimizing our forward loss-space matching objective bounds the divergence between the learned clean-sample distribution and the optimal reward-tilted clean-sample distribution. Our contribution can be summarized as follows:

- We propose `FCRM`, which frames diffusion RL post-training as a one-step GFlowNet matching problem in the forward loss-induced space. The resulting objective is forward consistent, naturally supports off-policy learning, and is directly compatible with few-step distilled video generators, making it suited for video generation.
- We establish a theoretical connection between loss-space matching and the original clean sample RL objective, clarifying when forward loss matching recovers the optimal KL-regularized RL solution on the clean sample distribution.
- Empirically, we develop a practical optimization recipe using a double-sampling surrogate for unbiased gradients, and validate the method on diffusion video RL post-training across online, replay, offline, and few-step settings.

2 Problem Setup

RL Training Goal. Let c denote a condition, such as a text prompt, and let $x_0 \in \mathbb{R}^D$ denote a clean sample. In our setting, x_0 may represent a video latent, but we use the generic notation x_0 for

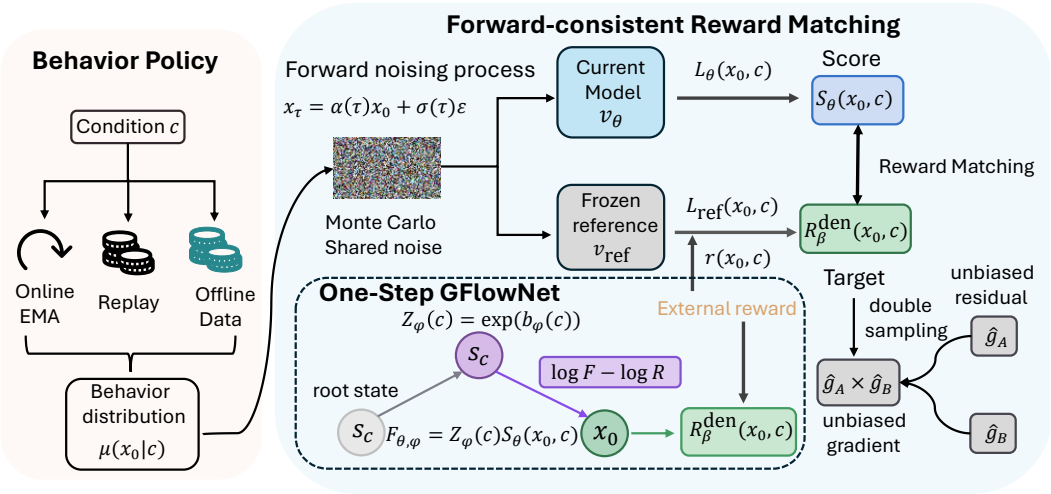

Figure 1: **Overview of** FCRM. **Left**: The off-policy formulation allows for flexible sampling from an online EMA model, replay buffers, and offline data to construct the behavior distribution. **Top**: Clean samples are perturbed using the standard forward noising process with shared Monte Carlo noise. **Bottom**: Denoising losses are converted into loss-induced scores. We match the edge flow from a condition-specific root state (s_c) to the terminal sample (x_0) against a target terminal reward. A double-sampling strategy is used to compute an unbiased gradient of the matching residual.

simplicity. A pretrained conditional diffusion model induces distribution $\pi_\theta(x_0 \mid c)$. Let $\pi_{\text{ref}} = \pi_{\theta_{\text{ref}}}$ denote a fixed reference model. Given a reward $r(x_0, c)$[1] and reward temperature β, our high-level goal is to improve expected reward under the generator:

$$\max_\theta \mathbb{E}_{x_0 \sim \pi_\theta(\cdot|c)}[r(x_0, c)] - \frac{1}{\beta} D_{\text{KL}}(\pi_\theta(\cdot \mid c) \,\|\, \pi_{\text{ref}}(\cdot \mid c)) \tag{1}$$

For each condition c, the corresponding optimal target is the reward-tilted distribution $\pi^\star_\beta(x_0 \mid c) \propto \pi_{\text{ref}}(x_0 \mid c) \exp(\beta r(x_0, c))$. The difficulty is that the exact clean-sample likelihood is generally unavailable for diffusion models. Rather than optimizing Eq. (1) directly, we define an analogous reward-tilted target in a loss-induced space built from the forward denoising loss and match the corresponding loss-space density. This preserves the forward denoising structure while avoiding explicit clean-sample or trajectory likelihood estimation.

Forward consistency. Let $\{\pi_{\tau|0}(\cdot \mid x_0)\}_{\tau \in [0,1]}$ denote the fixed forward noising process used in pretraining. This forward process induces the joint coupling $\pi_\theta(x_\tau, x_0 \mid c) = \pi_\theta(x_0 \mid c)\, \pi_{\tau|0}(x_\tau \mid x_0)$ and hence the time-τ marginal density path $\pi_{\theta,\tau}(x_\tau \mid c) = \int \pi_\theta(x_0 \mid c)\, \pi_{\tau|0}(x_\tau \mid x_0)\, dx_0$. We say that a post-training objective is *forward consistent* if its learning signal is defined through this fixed forward kernel and the induced density path $\{\pi_{\theta,\tau}\}_{\tau \in [0,1]}$ rather than through a solver-dependent reverse-time trajectory law. In continuous time, $\{\pi_{\theta,\tau}\}_{\tau \in [0,1]}$ evolve under the Fokker–Planck equation [35].

3 Method

3.1 Preliminaries

Forward Denoising Loss. The key object in our formulation is the standard forward denoising loss. This is the same primitive used in diffusion pretraining, which is why the resulting objective remains forward consistent. We begin from the standard forward noising construction:

$$x_\tau = \alpha(\tau)x_0 + \sigma(\tau)\epsilon, \qquad \tau \in [0, 1], \quad \epsilon \sim \mathcal{N}(0, I), \tag{2}$$

[1]We use advantage as a stabilization heuristic in implementation (see Appendix D). The theoretical target is defined using the reward r.

where $\alpha(\tau)$ and $\sigma(\tau)$ are noise schedule functions that define the signal-to-noise ratio at time τ. Let $v_\theta(x_\tau, \tau, c)$ denote the output of the neural network under v-prediction parameterization. Accordingly, let $u(\tau, x_0, \epsilon) = \alpha(\tau)\epsilon - \sigma(\tau)x_0$ denote the corresponding supervised v-prediction target. We define the per-sample forward denoising loss:

$$\mathcal{L}_\theta(x_0; c) = \mathbb{E}_{\tau,\epsilon}\Big[w(\tau)\, \|v_\theta(x_\tau, \tau, c) - u(\tau, x_0, \epsilon)\|_2^2\Big], \tag{3}$$

where $w(\tau)$ is a positive weighting function for the loss schedule (e.g., SNR weighting), τ is drawn from a fixed training-time distribution on $[0, 1]$ (e.g., uniform), and $\epsilon \sim \mathcal{N}(0, I)$. Throughout, $\mathcal{L}_{\text{ref}}(x_0; c)$ denotes the same quantity evaluated under the fixed reference model θ_{ref}. Our next step is to convert this denoising loss into a positive sample-wise score. This lets us define the learning problem in a forward loss-induced space, without introducing reverse-time trajectory likelihoods.

Loss-Induced Scores. We define a positive loss-induced score as $S_\theta(x_0, c) = \exp\Big(-\frac{1}{\gamma}\mathcal{L}_\theta(x_0; c)\Big)$. Large values of $S_\theta(x_0, c)$ correspond to samples that the current model denoises well, while poor denoising yields exponentially smaller scores. The scale γ controls how aggressively denoising-loss differences are converted into log-score differences. S_θ induces a normalized loss-space density: $q_\theta^{\text{den}}(x_0 \mid c) := \frac{S_\theta(x_0,c)}{\int S_\theta(x,c)\,dx}$. q_θ^{den} is, in general, not the diffusion-induced marginal distribution $\pi_\theta(x_0 \mid c)$. It is the density induced by the forward denoising quality. Our algorithm matches densities in this loss-induced space first and then, through the ELBO-consistency result (see Appendix E), transfers that solution back to a statement about the clean-sample distribution.

3.2 Loss-Space Reward Matching

With denoising quality converted into a positive score, reward alignment can be phrased as a one-step GFlowNet [3] terminal matching problem.

One-step GFlowNet View. For each fixed condition c, we consider a one-step DAG with root state s_c and terminal state for each clean sample x_0. The unique action from s_c is to emit a terminal sample x_0, so actions and terminal states are in one-to-one correspondence.

This viewpoint is natural for video generation, because the object that finally receives a reward is the *clean terminal sample* x_0, while the internal denoising path may be generated by *any black-box sampler*. A one-step GFlowNet view lets us separate the unnormalized target from its normalizer and yields a residual whose fixed point is pointwise in (x_0, c), which makes it off-policy compatible.

One-step GFlowNet Residual. Following the definition of GFlowNet, we define the root flow as $Z_\phi(c) = \exp(b_\phi(c))$, where $b_\phi(c)$ is a condition-dependent scalar parameterized by a separate neural network with weights ϕ. It acts as a learnable flow scale and log-normalizer which absorbs the unknown normalizing constant of the loss-induced score.

The edge flow from the root state s_c to the terminal state x_0 is defined as:

$$F_{\theta,\phi}(s_c \to x_0) = Z_\phi(c)\, S_\theta(x_0, c) = \exp\left(b_\phi(c) - \frac{1}{\gamma}\mathcal{L}_\theta(x_0; c)\right).$$

For a reward temperature $\beta > 0$, let the terminal reward be $R_\beta^{\text{den}}(x_0, c) = \exp\Big(\beta r(x_0, c) - \frac{1}{\gamma}\mathcal{L}_{\text{ref}}(x_0; c)\Big)$. This target is large when the sample both receives high external reward and low denoising loss under the reference model. The corresponding normalized loss-space target is $q_\beta^\star(x_0 \mid c) = \frac{R_\beta^{\text{den}}(x_0,c)}{Z_\beta^{\text{den}}(c)}$, where $Z_\beta^{\text{den}}(c) = \int R_\beta^{\text{den}}(x_0, c)\,dx_0$. With the above edge flow and terminal reward defined, we can now perform GFlowNet matching, a well-established GFlowNet training method to align flows with the reward. Since each terminal state x_0 has exactly one incoming edge and no outgoing edges, the standard GFlowNet detailed balance condition reduces to matching the edge flow directly to the terminal reward:

$$F_{\theta,\phi}(s_c \to x_0) = R_\beta^{\text{den}}(x_0, c).$$

Taking logs yields the residual:

$$
\begin{aligned}
g_{\theta,\phi}(x_0, c) &:= \log F_{\theta,\phi}(s_c \to x_0) - \log R_\beta^{\text{den}}(x_0, c) \\
&= b_\phi(c) - \beta r(x_0, c) + \frac{1}{\gamma}\big(\mathcal{L}_{\text{ref}}(x_0; c) - \mathcal{L}_\theta(x_0; c)\big).
\end{aligned} \tag{4}
$$

The object is a loss-space, one-step GFlowNet residual that matches a positive edge flow to a terminal reward. This construction turns diffusion RL into a one-step GFlowNet matching problem over clean samples, without requiring exact sample likelihoods or reverse-time trajectory likelihoods. Intuitively, Eq. (4) encourages the model to decrease denoising loss relative to the reference on high-reward samples. At any zero-residual solution, $S_\theta(x_0, c) = \exp(-b_\phi(c)) R_\beta^{\text{den}}(x_0, c)$. Therefore, after normalization over x_0, the learned loss-space density satisfies $q_\theta^{\text{den}}(x_0|c) = q_\beta^\star(x_0|c)$. The scalar $b_\phi(c)$ absorbs the target normalizer.

Although the algorithm above is defined entirely in the forward loss-induced space, it is not merely a heuristic surrogate detached from the original RL objective. In Appendix E, we show that, under the standard ELBO-consistency assumption for denoising losses, residual matching in loss space transfers to the clean-sample reward-tilted target $\pi_\beta^\star(x_0 \mid c) \propto \pi_{\text{ref}}(x_0 \mid c) \exp\big(\beta r(x_0, c)\big)$. When the loss-space residual is small and the ELBO-gap mismatch is controlled, the learned clean-sample distribution remains uniformly close to the optimal reward-tilted clean-sample distribution (see Proposition 3). This justifies the faithfulness of our GFlowNet formulation.

Off-Policy Compatibility. Let $\mu(\cdot \mid c)$ be a behavior distribution with adequate coverage of the high-reward region. We optimize:

$$
\mathcal{J}(\theta, \phi) = \mathbb{E}_{c \sim p(c),\, x_0 \sim \mu(\cdot|c)}\big[g_{\theta,\phi}(x_0, c)^2\big]. \tag{5}
$$

As in GFlowNet residual matching, the desired fixed point $g_{\theta,\phi}(x_0, c) = 0$ is pointwise in (x_0, c). In particular, any representable zero-residual solution is a global minimizer of Eq. (5) for any μ whose support covers the relevant region. In the realizable zero-residual limit, our target does not change with the behavior distribution μ. Thus the method is *naturally off-policy compatible and can exploit stale or offline samples*. In practice, the behavior distribution μ still affects the optimization via the variance and coverage. We therefore study EMA sampling (see Appendix D.4), replay, and fully offline data as a behavior distribution in the experiments. In Appendix D, we show that our method could be extended directly to few-step generators, which are often the practical choice for video generation.

Offline Data. As noted by Ye et al. [44], online policy optimization in video generation is notoriously vulnerable to reward hacking and the degradation of visual fidelity. To mitigate this, we incorporate offline video data primarily to anchor the learned distribution against reward exploitation. Concurrently, the incorporation of offline videos can reduce the computational overhead of online sampling. However, we view the primary benefit of incorporating offline videos as providing high-quality off-policy terminal samples that broaden the training support and stabilize reward optimization.

A key design choice in our approach is that offline videos are excluded from the group-relative advantage computation used for online samples. Instead, we decouple the online and offline streams into parallel FCRM objective branches. This formulation preserves the pointwise residual-matching nature of FCRM, allowing offline videos to act seamlessly as off-policy terminal samples. Importantly, we do not augment the objective with an auxiliary supervised denoising loss. Because offline data is optimized strictly through the `FCRM` residual, this integration functions as a *principled offline RL component* rather than a heuristic supervised fine-tuning (SFT) or data-regularization penalty. The offline branch implementation details can be found in Appendix D.2.

3.3 Monte Carlo Estimation and Practical Objective

The exact denoising loss $\mathcal{L}_\theta(x_0; c)$ in Eq. (3) is itself an expectation over (τ, ϵ). In practice, we estimate it using shared Monte Carlo samples.

Shared-noise Loss Estimator. We sample N_{mc} i.i.d. pairs $\{(\tau_j, \epsilon_j)\}_{j=1}^{N_{\mathrm{mc}}}$ and define

$$\widehat{\mathcal{L}}_\theta(x_0; c) = \frac{1}{N_{\mathrm{mc}}} \sum_{j=1}^{N_{\mathrm{mc}}} w(\tau_j) \, \|v_\theta(x_{\tau_j}, \tau_j, c) - u(\tau_j, x_0, \epsilon_j)\|_2^2. \tag{6}$$

We reuse the same (τ_j, ϵ_j) to estimate both $\widehat{\mathcal{L}}_\theta$ and $\widehat{\mathcal{L}}_{\theta_{\mathrm{ref}}}$. This common-random-number construction reduces the variance of the loss difference. We form the stochastic residual as: $\widehat{g}_{\theta,\phi}(x_0, c) = b_\phi(c) - \beta r(x_0, c) + \frac{1}{\gamma}\big(\widehat{\mathcal{L}}_{\mathrm{ref}}(x_0; c) - \widehat{\mathcal{L}}_\theta(x_0; c)\big)$.

Unbiased Gradient Estimation. The objective $\mathcal{J}(\theta, \phi)$ contains the square of an inner expectation, so replacing $g_{\theta,\phi}$ by a single Monte Carlo estimate yields a biased stochastic gradient (see Proposition 1). To avoid this bias, for each sample (x_0, c), we draw two conditionally independent noising batches, denoted A and B, and construct two unbiased residual estimates $\widehat{g}^A_{\theta,\phi}$ and $\widehat{g}^B_{\theta,\phi}$. We then use the double-sampling surrogate as the final objective:

$$\widehat{\mathcal{J}}_{\mathrm{plain}}(\theta, \phi) := \mathbb{E}_{c \sim p(c),\, x_0 \sim \mu(\cdot|c),\, A,B} \left[\widehat{g}^A_{\theta,\phi}(x_0, c)\, \widehat{g}^B_{\theta,\phi}(x_0, c)\right]. \tag{7}$$

Because $\widehat{g}^A_{\theta,\phi}(x_0, c)$ and $\widehat{g}^B_{\theta,\phi}(x_0, c)$ are conditionally independent and unbiased for $g_{\theta,\phi}(x_0, c)$, we have $\mathbb{E}\left[\widehat{g}^A_{\theta,\phi}(x_0, c)\, \widehat{g}^B_{\theta,\phi}(x_0, c) \,\middle|\, x_0, c\right] = g_{\theta,\phi}(x_0, c)^2$. In practice, we implement the corresponding unbiased gradient using the symmetrized stop-gradient surrogate: $\mathrm{sg}\big(\widehat{g}^A_{\theta,\phi}(x_0, c)\big)\widehat{g}^B_{\theta,\phi}(x_0, c) + \mathrm{sg}\big(\widehat{g}^B_{\theta,\phi}(x_0, c)\big)\widehat{g}^A_{\theta,\phi}(x_0, c)$, where $\mathrm{sg}(\cdot)$ denotes stop-gradient.

4 Related Work

RL Post-Training for Diffusion Models Early approaches [4, 13] to diffusion RL primarily formulated the reverse sampling trajectory as a multi-step Markov Decision Process (MDP). Recent advancements leverage Group Relative Policy Optimization (GRPO) [34] to align diffusion and flow-based models (e.g., Flow-GRPO [25]). To address the high variance and credit assignment challenges inherent in multi-step denoising, various trajectory structuring techniques (e.g., TempFlow-GRPO [15], Branch-GRPO [22], Chunk-GRPO [26], Dance-GRPO [43]) and dense reward stabilizers (e.g., Dense-GRPO [11], GARDO [14]) have been proposed. Despite these algorithmic improvements, backward-process RL inherently requires simulating full sampling trajectories during training. Most prior work in this line has focused on text-to-image models. Our emphasis is on video generation, where rollout cost is substantially higher.

To bypass the cost of backward-process RL, recent works have shifted to the forward process [10, 42, 48]. Advantage Weighted Matching [42] uses a forward denoising surrogate to estimate clean-sample likelihoods and performs advantage-weighted updates in that surrogate space. Similarly, DiffusionNFT [48] operates on the forward process, framing RL post-training as an implicit policy-improvement operator constructed from positive and negative samples. While our method also operates in the forward process, our distinction is that we define an explicit positive target distribution in loss space and optimize a pointwise residual whose definition does not depend on the behavior distribution. This makes our method fully compatible with off-policy learning and few-step generators. The replay and offline-data reuse are especially natural in our formulation, whereas prior forward-space methods have been presented and evaluated primarily in near-on-policy settings.

Implicit RL Post-Training for Diffusion Models Parallel to explicit-reward RL, implicit RL methods attempt to align diffusion models from preference data. Diffusion-DPO [37] adapts Direct Preference Optimization [30] to diffusion models by formulating the objective over the implicit reward defined by the diffusion model's score function. GPO [9] and DGPO [27] extend this to group-level preferences, while GDRO [40] reframes the log-likelihood ratio into a cross-entropy training objective. These methods optimize empirical preference-learning objectives defined over comparison data or sampled preference groups from the implicit reward, rather than matching a fixed explicit reward-tilted target distribution. Consequently, changing the underlying pairwise or groupwise data distribution generally changes the optimization problem [2, 36]. This differs from the off-policy setting studied in this paper, where the target is fixed by an explicit reward function and reference model, while the behavior distribution affects only coverage and variance.

Table 1: Main Video-Level Evaluation Results on Wan 2.1-1.3B. We compare FCRM against baselines on VBench for fine-grained video evaluation and on video-level Latent Reward for human-preference evaluation. **Bold** indicates the best performance, and underline indicates the second best. GPU-h indicates the total training time. We evaluate the video score on six dimensions using VBench.

Method	GPU-h↓	Video-Level		VBench Comprehensive Evaluation					
		Latent Reward↑	VBench Avg ↑	Temporal Consistency		Motion Quality		Visual Quality	
				Subject	Background	Smoothness	Dynamic	Aesthetic	Imaging
Wan 2.1-1.3B (4 NFE)	–	2.39	79.30	96.42	95.76	98.94	54.40	61.48	68.81
Flow-GRPO (28 NFE)	57.0	2.07	78.09	95.39	95.77	98.43	51.40	61.55	66.01
AWM (4 NFE)	18.6	**4.12**	80.04	**97.12**	95.80	98.77	54.80	**63.54**	70.23
DiffusionNFT (4 NFE)	18.0	3.10	76.71	96.31	95.69	**99.04**	46.91	55.58	67.04
FCRM	19.2	4.03	**80.63**	96.75	**95.91**	98.99	**57.00**	62.87	**72.26**

5 Experiments

5.1 Experiment Setup

Evaluation Setup. We implement our proposed method within the Flow-Factory framework [29]. For training, we randomly sample 20,000 text prompts from VidProM [39]. For evaluation, we reserve a disjoint set of 500 VidProM prompts. To comprehensively evaluate the quality and alignment of our generated videos, we employ a diverse suite of metrics. Specifically, we use VBench [18] for fine-grained, multi-dimensional video generation evaluation. Additionally, we assess frame-level visual quality, text-alignment, and human preference using established metrics including PickScore [21], CLIPScore [19], HPSv2.1 [41], and Aesthetic scores [33]. All videos are generated at resolution 480×832 with 41 frames using the 4-NFE few-step Wan 2.1-1.3B generator [46].

Implementation Setup. We evaluate our approach using the Wan 2.1-1.3B video generation model. A key practical advantage of our off-policy formulation is its flexibility in data collection. Consequently, our training leverages a hybrid data mixture: online samples generated via an EMA of the behavior policy, mixed with offline data to regularize the training dynamics and mitigate reward hacking. We include the implementation details for offline video incorporation and the hyperparameter details in Appendix D.

Reward Computation. To avoid costly video decoding during the RL training loop, our default implementation uses a latent-space video reward model. Given a generated terminal latent x_0 and prompt c, the reward model directly outputs a scalar score $r(x_0, c)$ in the generator's latent space. The reward model is trained separately, kept frozen during FCRM training, and used only as a detached evaluator. This latent-space design substantially reduces training cost because it avoids VAE decoding for every sampled video. It also matches the latent-domain training setup of modern video generators. The latent reward model uses the frozen Wan 2.1-1.3B video diffusion backbone as a noise-aware feature extractor. Intermediate spatio-temporal DiT features are extracted from the latent stream and combined with the model's text features. A lightweight query-based aggregation head then cross-attends to these visual-text tokens and maps the aggregated representation to a scalar reward. We compare against pixel-space reward computation in the ablation study, where generated latents are decoded to RGB videos before being scored.

Data Collection. To test our off-policy capabilities, we evaluate FCRM under several behavior distributions: fully on-policy sampling, hybrid EMA sampling, replay-buffer sampling, and offline video data. For EMA sampling, videos are generated from an exponential moving average of the current model. For replay, we maintain a FIFO buffer of previously generated samples and reuse them as off-policy data. The replay buffer is capped at 20000. The reward is computed once when the sample is inserted into the buffer and reused during subsequent training. At training time, the replay samples are drawn uniformly from the current buffer. Once inserted, a sample remains available until removed by FIFO eviction. For offline training, videos are encoded into the generator latent space and treated as fixed off-policy samples.

Table 2: Frame-Level Human Preference Evaluation. Comparison of our proposed FCRM against baselines on standard image-level preference metrics.

Method	Steps	PickScore ↑	HPSv2.1 ↑	CLIPScore ↑	Aesthetic ↑	Avg ↑
Wan 2.1-1.3B	few-step	20.65	26.37	28.27	6.04	20.33
Flow-GRPO	multi-step	20.32	24.63	**29.16**	5.93	20.01
AWM	few-step	**21.20**	27.10	28.49	**6.34**	20.78
DiffusionNFT	few-step	20.37	26.93	28.20	5.69	20.30
`FCRM`	few-step	20.99	**27.38**	28.65	6.27	**20.82**

5.2 Main Results

Table 1 reports the video-level evaluation results on Wan 2.1-1.3B. Compared to the pretrained few-step generator, FCRM improves the latent reward from 2.39 to 4.03 and improves the VBench average from 79.30 to 80.63. Compared to AWM and DiffusionNFT, FCRM achieves the best VBench average and the best image quality score, while remaining competitive in temporal consistency and visual quality. These results indicate that FCRM improves reward alignment without sacrificing the standard video-quality metrics measured by VBench. Table 2 reports the frame-level evaluation. FCRM achieves the highest average score and obtains the best HPSv2.1 and CLIPScore. This suggests that the gains from FCRM are not limited to the latent reward model used during training, but also transfer to independent frame-level preference alignment metrics.

Table 3: Comprehensive Ablation Study on FCRM Components. We evaluate the impact of reward formulation, optimization space, data collection strategies, and generator types.

Configuration	Latent Reward ↑	VBench Avg ↑	HPSv2.1 ↑	Aesthetic ↑	PickScore ↑
Wan 2.1-1.3B (4 NFE)	2.39	79.30	26.37	6.04	20.65
`FCRM (Default)`	**4.03**	**80.63**	27.38	**6.27**	20.99
Reward Target					
w/ Reward	2.56	79.20	26.15	6.08	20.68
Reward Space					
w/ Pixel-Space Reward	2.47	77.71	26.14	6.20	20.91
Data Collection Strategy					
Fully On-policy	3.27	79.18	26.65	6.13	20.41
Fully Offline	2.98	79.31	26.49	6.17	20.88
Replay Filter					
Positive-only Replay	3.96	78.44	26.59	6.10	**21.11**
Unfiltered Replay	3.82	80.10	**28.57**	6.18	21.07
Negative-only Replay	2.75	76.43	25.56	5.92	20.04
Generator Type					
Multi-step Model (28 NFE)	2.07	78.09	24.63	5.93	20.32

5.3 Ablation Study

Detached normalized advantage vs reward. We compare the reward objective with the practical detached group-normalized advantage used for stable optimization. As illustrated in Fig. 2, when optimizing the reward directly, the matching loss continuously decreases, yet the latent reward mean fails to improve. This indicates that the model easily minimizes the residual without meaningfully shifting the policy towards high-reward regions. In contrast, the normalized advantage provides a robust relative learning signal, effectively driving reward improvement.

KL Divergence Penalty. Following Flow-GRPO [25], we explored adding an explicit KL penalty loss to the FCRM objective with varying coefficients $\{0, 0.01, 1\}$. We observe that tuning this penalty is notoriously difficult. As in Fig. 2, a high coefficient overly restricts the policy and punishes reward learning, whereas a low coefficient loses its regularizing effect as training progresses. Instead of relying on a fragile KL penalty, `FCRM` enables a new paradigm of data-level regularization. By natively accommodating off-policy and offline data, FCRM anchors the learned distribution to high-quality

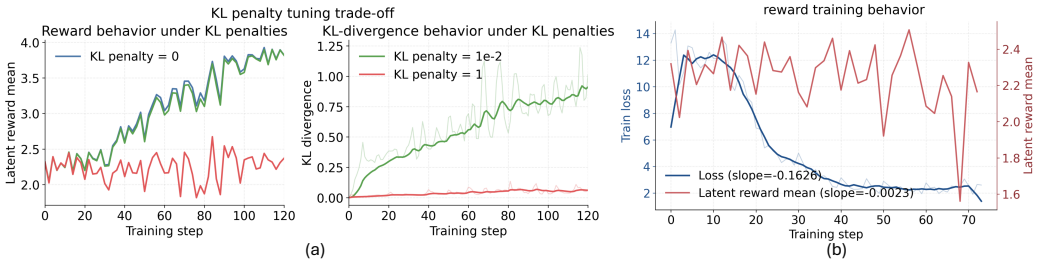

Figure 2: **Training dynamics of FCRM under different optimization settings.** (a) The latent reward mean over training steps when applying different KL divergence penalty coefficients $\{0, 0.01, 1\}$. (b) The training loss and latent reward mean over training steps when optimizing the reward directly without group-relative advantage normalization.

data, naturally stabilizing the optimization and preventing reward hacking without the need for sensitive KL tuning.

Lightness Reward. We train FCRM with a simple dense reward equal to the mean frame luminance and visualize four prompts at training steps 0, 80, and 160. The training is conducted with a fully-offline and off-policy setting. We only use the offline MixKit dataset [23] without online generated videos. As in Appendix Fig. 3, the overlaid luma_mean increases consistently with training, while the initial visual fidelity and prompt alignment is maintained.

Replay-buffer composition. Within the replay setting, we ablate the composition of the replay buffer by retaining only positive-advantage samples, only negative-advantage samples, or all samples without filtering. As shown in Table 3 and Fig. 7, the negative-only replay performs the worst in all replay variants, with a latent reward of 2.75 and a VBench average of 76.43. Qualitatively, the negative-only replay also produces severe artifacts as the training progresses. This is consistent with the fact that negative-advantage samples induce a one-sided suppression signal in the loss-induced score space. This behavior is analogous to the "squeezing effect" [31] analyzed in the learning dynamics of LLMs finetuning, where repeatedly applying negative update signals to already low-preference regions can shift the probability mass to uncontrolled modes.

The positive only and unfiltered replay strategies also did not achieve notable reward improvements or effective mitigation of reward hacking. Replay improves sample reuse, but the replay buffer is still populated by model-generated samples. As reward maximization progresses, generated samples can exploit blind spots of the reward model. Once such samples enter the replay buffer, their artifacts may be repeatedly reinforced during subsequent updates. This is especially problematic for learned video reward models, which are generally imperfect. Therefore, replay should be viewed as a sample-efficiency mechanism rather than a sufficient anti-hacking mechanism. In our default setting, we instead use hybrid online–offline training so that high-quality offline videos provide a stronger data-level anchor.

Pixel-space Reward. We compare pixel-space rewards versus latent-space rewards. We adopt PickScore as the pixel-space reward model, following [29]. As shown in Table 3, optimizing with a pixel-space reward yields an inferior performance in most metrics compared to our default latent-space reward. In contrast, the latent-space reward aligns better with the generator's native representation, while bypassing the prohibitive computational cost of VAE decoding during the RL loop.

6 Conclusion

We introduced FCRM, a forward-consistent off-policy RL post-training method for video generation. By converting the forward denoising loss into a loss-induced score and matching it to a reward-tilted target through a one-step GFlowNet residual, FCRM is naturally compatible with replay buffers, offline data, and few-step distilled generators. We further provided a double-sampling estimator for unbiased residual-gradient estimation and established a theoretical connection between loss-space matching and KL-regularized clean-sample reward optimization. Empirically, we conducted experiments using Wan base models and tested on multiple video evaluation benchmarks. FCRM provides an effective recipe for efficient video RL post-training across online, replay, offline, and few-step settings.

References

[1] Robert A Adams and John JF Fournier. *Sobolev spaces*, volume 140. Elsevier, 2003. (Cited on page 19)

[2] Mohammad Gheshlaghi Azar, Zhaohan Daniel Guo, Bilal Piot, Remi Munos, Mark Rowland, Michal Valko, and Daniele Calandriello. A general theoretical paradigm to understand learning from human preferences. In *International Conference on Artificial Intelligence and Statistics*, pages 4447–4455. PMLR, 2024. (Cited on page 6)

[3] Emmanuel Bengio, Moksh Jain, Maksym Korablyov, Doina Precup, and Yoshua Bengio. Flow network based generative models for non-iterative diverse candidate generation. *Advances in neural information processing systems*, 34:27381–27394, 2021. (Cited on pages 2 and 4)

[4] 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*. (Cited on pages 1 and 6)

[5] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023. (Cited on pages 1 and 14)

[6] Haim Brezis and Haim Brézis. *Functional analysis, Sobolev spaces and partial differential equations*, volume 2. Springer, 2011. (Cited on page 19)

[7] Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Leo Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, et al. Video generation models as world simulators. *OpenAI Blog*, 1(8):1, 2024. (Cited on pages 1 and 14)

[8] Haoxin Chen, Menghan Xia, Yingqing He, Yong Zhang, Xiaodong Cun, Shaoshu Yang, Jinbo Xing, Yaofang Liu, Qifeng Chen, Xintao Wang, et al. Videocrafter1: Open diffusion models for high-quality video generation. *arXiv preprint arXiv:2310.19512*, 2023. (Cited on pages 1 and 14)

[9] Renjie Chen, Wenfeng Lin, Yichen Zhang, Jiangchuan Wei, Boyuan Liu, Chao Feng, Jiao Ran, and Mingyu Guo. Towards self-improvement of diffusion models via group preference optimization. *arXiv preprint arXiv:2505.11070*, 2025. (Cited on page 6)

[10] Jaemoo Choi, Yuchen Zhu, Wei Guo, Petr Molodyk, Bo Yuan, Jinbin Bai, Yi Xin, Molei Tao, and Yongxin Chen. Rethinking the design space of reinforcement learning for diffusion models: On the importance of likelihood estimation beyond loss design. *arXiv preprint arXiv:2602.04663*, 2026. (Cited on page 6)

[11] Haoyou Deng, Keyu Yan, Chaojie Mao, Xiang Wang, Yu Liu, Changxin Gao, and Nong Sang. Densegrpo: From sparse to dense reward for flow matching model alignment. *arXiv preprint arXiv:2601.20218*, 2026. (Cited on page 6)

[12] Zihan Ding, Chi Jin, Difan Liu, Haitian Zheng, Krishna Kumar Singh, Qiang Zhang, Yan Kang, Zhe Lin, and Yuchen Liu. Dollar: Few-step video generation via distillation and latent reward optimization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17961–17971, 2025. (Cited on page 2)

[13] Ying Fan, Olivia Watkins, Yuqing Du, Hao Liu, Moonkyung Ryu, Craig Boutilier, Pieter Abbeel, Mohammad Ghavamzadeh, Kangwook Lee, and Kimin Lee. Dpok: Reinforcement learning for fine-tuning text-to-image diffusion models. *Advances in Neural Information Processing Systems*, 36:79858–79885, 2023. (Cited on pages 1 and 6)

[14] Haoran He, Yuxiao Ye, Jie Liu, Jiajun Liang, Zhiyong Wang, Ziyang Yuan, Xintao Wang, Hangyu Mao, Pengfei Wan, and Ling Pan. Gardo: Reinforcing diffusion models without reward hacking. *arXiv preprint arXiv:2512.24138*, 2025. (Cited on page 6)

[15] Xiaoxuan He, Siming Fu, Yuke Zhao, Wanli Li, Jian Yang, Dacheng Yin, Fengyun Rao, and Bo Zhang. Tempflow-grpo: When timing matters for grpo in flow models. *arXiv preprint arXiv:2508.04324*, 2025. (Cited on page 6)

[16] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. (Cited on page 1)

[17] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022. (Cited on page 14)

[18] 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. (Cited on page 7)

[19] Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. Openclip, July 2021. URL https://doi.org/10.5281/zenodo.5143773. If you use this software, please cite it as below. (Cited on page 7)

[20] Dengyang Jiang, Dongyang Liu, Zanyi Wang, Qilong Wu, Liuzhuozheng Li, Hengzhuang Li, Xin Jin, David Liu, Changsheng Lu, Zhen Li, et al. Distribution matching distillation meets reinforcement learning. *arXiv preprint arXiv:2511.13649*, 2025. (Cited on page 2)

[21] Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. Pick-a-pic: An open dataset of user preferences for text-to-image generation. *Advances in neural information processing systems*, 36:36652–36663, 2023. (Cited on page 7)

[22] Yuming Li, Yikai Wang, Yuying Zhu, Zhongyu Zhao, Ming Lu, Qi She, and Shanghang Zhang. Branchgrpo: Stable and efficient grpo with structured branching in diffusion models. *arXiv preprint arXiv:2509.06040*, 2025. (Cited on page 6)

[23] Bin Lin, Yunyang Ge, Xinhua Cheng, Zongjian Li, Bin Zhu, Shaodong Wang, Xianyi He, Yang Ye, Shenghai Yuan, Liuhan Chen, et al. Open-sora plan: Open-source large video generation model. *arXiv preprint arXiv:2412.00131*, 2024. (Cited on page 9)

[24] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations*. (Cited on pages 1 and 14)

[25] Jie Liu, Gongye Liu, Jiajun Liang, Yangguang Li, Jiaheng Liu, Xintao Wang, Pengfei Wan, Di ZHANG, and Wanli Ouyang. Flow-grpo: Training flow matching models via online rl. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*. (Cited on pages 1, 6 and 8)

[26] Yifu Luo, Penghui Du, Bo Li, Sinan Du, Tiantian Zhang, Yongzhe Chang, Kai Wu, Kun Gai, and Xueqian Wang. Sample by step, optimize by chunk: Chunk-level grpo for text-to-image generation. *arXiv preprint arXiv:2510.21583*, 2025. (Cited on page 6)

[27] Yihong Luo, Tianyang Hu, and Jing Tang. Reinforcing diffusion models by direct group preference optimization. *arXiv preprint arXiv:2510.08425*, 2025. (Cited on page 6)

[28] 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. (Cited on page 14)

[29] Bowen Ping, Chengyou Jia, Minnan Luo, Hangwei Qian, and Ivor Tsang. Flow-factory: A unified framework for reinforcement learning in flow-matching models. *arXiv preprint arXiv:2602.12529*, 2026. URL https://arxiv.org/abs/2602.12529. (Cited on pages 7 and 9)

[30] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741, 2023. (Cited on page 6)

[31] Yi Ren and Danica J. Sutherland. Learning dynamics of LLM finetuning. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*, 2025. (Cited on page 9)

[32] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. *arXiv preprint arXiv:2202.00512*, 2022. (Cited on page 2)

[33] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade W Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa R Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. LAION-5b: An open large-scale dataset for training next generation image-text models. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022. URL https://openreview.net/forum?id=M3Y74vmsMcY. (Cited on page 7)

[34] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024. (Cited on page 6)

[35] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020. (Cited on pages 1 and 3)

[36] Fahim Tajwar, Anikait Singh, Archit Sharma, Rafael Rafailov, Jeff Schneider, Tengyang Xie, Stefano Ermon, Chelsea Finn, and Aviral Kumar. Preference fine-tuning of llms should leverage suboptimal, on-policy data. In *International Conference on Machine Learning*, pages 47441–47474. PMLR, 2024. (Cited on page 6)

[37] Bram Wallace, Meihua Dang, Rafael Rafailov, Linqi Zhou, Aaron Lou, Senthil Purushwalkam, Stefano Ermon, Caiming Xiong, Shafiq Joty, and Nikhil Naik. Diffusion model alignment using direct preference optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8228–8238, 2024. (Cited on page 6)

[38] Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wente Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025. (Cited on pages 1 and 14)

[39] Wenhao Wang and Yi Yang. Vidprom: A million-scale real prompt-gallery dataset for text-to-video diffusion models. 2024. URL https://openreview.net/forum?id=pYNl76onJL. (Cited on page 7)

[40] Yiyang Wang, Xi Chen, Xiaogang Xu, Yu Liu, and Hengshuang Zhao. Gdro: Group-level reward post-training suitable for diffusion models. *arXiv preprint arXiv:2601.02036*, 2026. (Cited on page 6)

[41] Xiaoshi Wu, Yiming Hao, Keqiang Sun, Yixiong Chen, Feng Zhu, Rui Zhao, and Hongsheng Li. Human preference score v2: A solid benchmark for evaluating human preferences of text-to-image synthesis. *arXiv preprint arXiv:2306.09341*, 2023. (Cited on page 7)

[42] Shuchen Xue, Chongjian Ge, Shilong Zhang, Yichen Li, and Zhi-Ming Ma. Advantage weighted matching: Aligning rl with pretraining in diffusion models. *arXiv preprint arXiv:2509.25050*, 2025. (Cited on pages 1 and 6)

[43] Zeyue Xue, Jie Wu, Yu Gao, Fangyuan Kong, Lingting Zhu, Mengzhao Chen, Zhiheng Liu, Wei Liu, Qiushan Guo, Weilin Huang, et al. Dancegrpo: Unleashing grpo on visual generation. *arXiv preprint arXiv:2505.07818*, 2025. (Cited on page 6)

[44] Haotian Ye, Kaiwen Zheng, Jiashu Xu, Puheng Li, Huayu Chen, Jiaqi Han, Sheng Liu, Qinsheng Zhang, Hanzi Mao, Zekun Hao, et al. Data-regularized reinforcement learning for diffusion models at scale. *arXiv preprint arXiv:2512.04332*, 2025. (Cited on page 5)

[45] Tianwei Yin, Michaël Gharbi, Taesung Park, Richard Zhang, Eli Shechtman, Fredo Durand, and William T Freeman. Improved distribution matching distillation for fast image synthesis. *Advances in neural information processing systems*, 37:47455–47487, 2024. (Cited on page 2)

[46] Tianwei Yin, Qiang Zhang, Richard Zhang, William T Freeman, Fredo Durand, Eli Shechtman, and Xun Huang. From slow bidirectional to fast autoregressive video diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22963–22974, 2025. (Cited on page 7)

[47] Haiyu Zhang, Xinyuan Chen, Yaohui Wang, Xihui Liu, Yunhong Wang, and Yu Qiao. Accvideo: Accelerating video diffusion model with synthetic dataset. *arXiv preprint arXiv:2503.19462*, 2025. (Cited on page 2)

[48] Kaiwen Zheng, Huayu Chen, Haotian Ye, Haoxiang Wang, Qinsheng Zhang, Kai Jiang, Hang Su, Stefano Ermon, Jun Zhu, and Ming-Yu Liu. Diffusionnft: Online diffusion reinforcement with forward process. *arXiv preprint arXiv:2509.16117*, 2025. (Cited on pages 1 and 6)

A Broader Impacts

Our work presents an efficient approach to align large-scale video generation models with human preferences and external reward signals. A primary positive impact of FCRM is its significant reduction in the computational resources required for RL post-training. By bypassing the need for expensive multi-step reverse trajectory rollouts, our method lowers the barrier to entry for fine-tuning foundational video models. This democratization allows smaller research labs and organizations to align models without requiring massive compute clusters, while also contributing to a reduction in the carbon footprint associated with training large-scale AI systems.

As with any algorithmic advancement that improves the capabilities of generative AI, there are general, dual-use implications. Enhancing the visual quality and prompt-faithfulness of video models could marginally lower the effort required for malicious actors to generate misleading synthetic media or misinformation. Furthermore, because FCRM is designed to efficiently optimize a given reward signal, it is susceptible to "garbage in, garbage out" dynamics. If the external reward model contains societal biases, the fine-tuned video generator may inadvertently reflect or amplify those biases.

However, because FCRM is a general-purpose optimization algorithm rather than a standalone consumer application, these risks are not unique to our method. They are best mitigated at the deployment and application levels through responsible reward model design, strict usage guidelines, and the integration of synthetic media provenance techniques, such as robust digital watermarking.

B Future Works

In this work, we introduced FCRM, a forward-consistent, off-policy RL framework for video generation. By framing diffusion RL as a one-step GFlowNet matching problem in the forward loss-induced space, FCRM bypasses the prohibitive costs of multi-step reverse trajectory unrolling. Furthermore, it seamlessly accommodates few-step distilled generators, hybrid offline samples, and replay buffers, making it highly practical for modern video generation pipelines. FCRM provides a highly efficient alternative to traditional trajectory-based RL, and it also opens several promising avenues for future research. Although our objective is compatible with the off-policy, which means that the target residual remains consistent regardless of the behavior policy, the choice of data collection still influences the variance of the optimization process. Future work could explore more sophisticated exploration strategies to further accelerate convergence. FCRM achieves massive computational savings by eliminating multi-step sampling during training. However, to maintain an unbiased gradient estimator, our current practical implementation utilizes a double-sampling surrogate, which requires two forward passes per training step. An interesting direction for future research would be to investigate variance-reduced single-sample estimators or memory-efficient approximations to further reduce the memory footprint during fine-tuning.

C Additional Related Work

Video Generation Models Video generation has been advanced by diffusion models and flow matching techniques. Early methods extended image diffusion priors to the temporal domain, enabling high-fidelity video generation [5, 8, 17]. More recently, Flow Matching [24] and Diffusion Transformers [28] have emerged as highly scalable methods for modeling video distributions. State-of-the-art models include Sora [7] and Wan 2.1 [38]. These models are primarily optimized for data likelihood during pretraining and do not explicitly optimize for downstream human preferences, aesthetic quality, or prompt alignment. Our work studies RL post-training on top of the pretrained video generators with an emphasis on few-step distilled models and efficient, off-policy training, as video generation is increasingly computationally heavy.

D Implementation Details

D.1 Architectural Design of b_ϕ

The log-normalizer is implemented as a condition-dependent scalar network, rather than a fixed global constant. It takes the frozen text-encoder prompt embedding as input, mean-pools token

embeddings when the input is sequence-shaped, and passes the resulting condition vector through a lightweight MLP with SiLU activations and hidden size as 256 by default. The final linear layer is zero-initialized, so the normalizer initially contributes no offset and is learned from the residual.

D.2 FCRM with Offline Data

This section describes the hybrid online–offline training recipe used in our experiments. The key design choice is that offline videos are not inserted into the same group-relative advantage computation as generated videos. Instead, we keep the online generated-sample branch and the offline video branch as two separate FCRM objectives. This preserves the pointwise residual-matching form of FCRM while allowing offline videos to serve as off-policy terminal samples.

Online Advantage Calculation. In minibatch optimization, when a group $\{x_0^{(i)}\}_{i=1}^{G} \sim \pi_\theta(\cdot \mid c)$ is drawn for the same condition c, one may form the detached group-relative normalized advantage $\hat{A}^{(i)} = \frac{r(x_0^{(i)},c) - \frac{1}{G}\sum_{j=1}^{G} r(x_0^{(j)},c)}{\sqrt{\frac{1}{G}\sum_{j=1}^{G}\left(r(x_0^{(j)},c) - \frac{1}{G}\sum_{k=1}^{G} r(x_0^{(k)},c)\right)^2}}$. The group statistics are treated as detached constants when computing gradients.

Offline Video Branch. Let $\mathcal{D}_{\text{offline}}$ denote the offline video dataset with captions. Each sample consists of an offline video latent and its caption. At each epoch, we sample N_{offline} unique MixKit caption-video pairs. The video is encoded into the generator latent space and is treated as an off-policy terminal sample. Unlike the online branch, no generated videos are required to form the residual. For each sample, we compute the reward. Since each caption is typically paired with only one video, group-relative normalization within a caption is not available. We therefore use source-level reward normalization over the offline branch. At epoch e, we compute the batch mean $\widehat{\mu}_e$ and second moment $\widehat{m}_e$. We maintain exponential moving average statistics and define $\sigma_e = \sqrt{\max\left(m_e - \mu_e^2, \sigma_{\min}^2\right)}$. The detached offline normalized advantage is then computed with GRPO-style reward normalization.

Offline FCRM residual. The offline branch uses the same FCRM residual-matching principle as the online branch. The only difference is that the outer samples are videos drawn from $\mathcal{D}_{\text{offline}}$. The offline residual uses a source-level scalar log-normalizer b_{offline}. We use a source-level normalizer because each caption has only one video. An unconstrained caption-dependent normalizer trained on this branch could otherwise absorb much of the single-sample residual without producing a useful generator update. As an alternative implementation, we also consider using the caption-dependent normalizer from the online branch while stopping its gradient on offline samples. This preserves condition-dependent normalizer values while preventing the offline branch from being solved by updating b_ϕ alone. We use the same double-sampling residual estimator used in the main FCRM objective.

Signed versus nonnegative offline advantages. The signed offline advantage corresponds to a reward-weighted offline RL interpretation. High-reward offline videos are assigned positive advantages, while lower-reward offline videos receive negative advantages relative to the reward baseline. We also consider a conservative nonnegative variant:

$$\widehat{A}^{\text{offline},+} = \text{clip}\left(\frac{r - \mu_e}{\sigma_e + \epsilon} + a_0, 0, A_{\max}\right), \tag{1}$$

where $a_0 \geq 0$ is an optional offset. We set $a_0 = 1.0$ in our experiments. This variant treats videos as positive offline anchors and avoids assigning negative advantages to offline video samples.

Mixed objective. The final hybrid objective combines the online generated-sample and the offline FCRM branch with λ_{offline} to control the relative strength of the offline branch. In our implementation, the number of offline samples per epoch is chosen to match the number of online generated samples.

Pure offline setting. We also evaluate a pure-offline variant. The training uses only offline video-caption pairs. Equivalently, the behavior distribution is the empirical distribution over offline video-caption pairs. This gives a fully offline FCRM update. No generated training rollouts are used, but the model is still optimized by reward-matching through the FCRM residual.

Table 4: Hyperparameters for `FCRM`.

Hyperparameter	Value
Reward temperature β	`1`
Loss-score temperature γ	`1`
Group size G	`32`
N_{mc}	`1`
Timestep distribution $p(\tau)$	`Uniform`
Loss weight $w(\tau)$	`Uniform`
Sampling steps	`[1000, 757, 522]`
Replay buffer size	`20000`
Optimizer	`AdamW`
Learning rate for θ	`1.0e-4`
Learning rate for ϕ	`1.0e-3`
Weight decay	`1.0e-4`
Gradient clipping	`1.0`
Batch size	`768`
EMA decay η	`0.9`
Hardware	`4*NVIDIA H200`

D.3 Extension to Few-step Generators

Extension to a Distribution Matching Distillation (DMD)-distilled model. The compatibility with few-step generators is a central practical advantage of our formulation. Suppose now that the pretrained generator is a K-step distilled model with a fixed timestep set $\mathcal{T}_K = \{\tau_1, \ldots, \tau_K\} \subset [0, 1]$. We keep the notation $\pi_\theta(x_0 \mid c)$ for the terminal distribution induced by running this K-step sampler. Since the distilled model is queried only on $\mathcal{T}_K$, we replace Eq. (3) by the schedule-aware denoising loss:

$$\mathcal{L}_\theta^{(K)}(x_0; c) = \mathbb{E}_{\tau \sim p_{\mathcal{T}_K}, \epsilon}\Big[w(\tau)\, \|v_\theta(x_\tau, \tau, c) - u(\tau, x_0, \epsilon)\|_2^2\Big],$$

where $p_{\mathcal{T}_K}$ is any distribution supported on $\mathcal{T}_K$ (e.g. uniform). Let $\mathcal{L}_{\mathrm{ref}}^{(K)}(x_0; c) = \mathcal{L}_{\theta_{\mathrm{ref}}}^{(K)}(x_0; c)$. Accordingly, the objective is replaced by

$$g_{\theta,\phi}^{(K)}(x_0, c) = b_\phi(c) - \beta r(x_0, c) + \frac{1}{\gamma}\big(\mathcal{L}_{\mathrm{ref}}^{(K)}(x_0; c) - \mathcal{L}_\theta^{(K)}(x_0; c)\big).$$

The shared-noise Monte Carlo estimator in Eq. (6) is modified only by drawing timesteps from $p_{\mathcal{T}_K}$. with the same common-random-number reuse between θ and θ_{ref}.

D.4 EMA Sampling

In particular, online fine-tuning uses outer samples $x_0 \sim \mu(\cdot \mid c)$, where in the distilled K-step setting the behavior policy μ may be chosen either as a frozen copy of the current K-step distilled sampler or the EMA weights,

$$\mu = \pi_{\theta^{\mathrm{old}}}, \qquad \theta^{\mathrm{old}} \leftarrow \eta\, \theta^{\mathrm{old}} + (1 - \eta)\, \theta, \qquad \eta \in [0, 1).$$

E Theoretical Results

E.1 Unbiased Gradient Estimation

Proposition 1 (Bias of the Naive Squared Estimator)**.** *Fix c, and let $x_0 \sim \mu(\cdot \mid c)$. Suppose $\widehat{g}_{\theta,\phi}(x_0, c; \tau, \epsilon)$ is an unbiased single-sample Monte Carlo estimator of $g_{\theta,\phi}(x_0, c)$, i.e., $\mathbb{E}_{\tau,\epsilon}[\widehat{g}_{\theta,\phi}(x_0, c; \tau, \epsilon) \mid x_0, c] = g_{\theta,\phi}(x_0, c)$. Define the true objective $\mathcal{J}(\theta, \phi) = \mathbb{E}_{x_0 \sim \mu(\cdot|c)}[g_{\theta,\phi}(x_0, c)^2]$, and the single-sample naive estimator $\widehat{\mathcal{J}}_{\mathrm{naive}}(\theta, \phi) = \widehat{g}_{\theta,\phi}(x_0, c; \tau, \epsilon)^2$. Then:*

$$\mathbb{E}_{x_0,\tau,\epsilon}[\widehat{\mathcal{J}}_{\mathrm{naive}}(\theta, \phi)] = \mathcal{J}(\theta, \phi) + \mathbb{E}_{x_0 \sim \mu(\cdot|c)}\left[\mathrm{Var}_{\tau,\epsilon}\left(\widehat{g}_{\theta,\phi}(x_0, c; \tau, \epsilon) \mid x_0, c\right)\right].$$

Hence $\widehat{\mathcal{J}}_{\mathrm{naive}}$ is upward biased whenever the Monte Carlo variance is nonzero.

Proof. For fixed (x_0, c), the variance identity gives

$$\mathbb{E}_{\tau,\epsilon}[\widehat{g}_{\theta,\phi}(x_0, c; \tau, \epsilon)^2 \mid x_0, c] = (\mathbb{E}_{\tau,\epsilon}[\widehat{g}_{\theta,\phi}(x_0, c; \tau, \epsilon) \mid x_0, c])^2 + \mathrm{Var}_{\tau,\epsilon}\left(\widehat{g}_{\theta,\phi}(x_0, c; \tau, \epsilon) \mid x_0, c\right).$$

By unbiasedness,

$$\mathbb{E}_{\tau,\epsilon}[\widehat{g}_{\theta,\phi}(x_0, c; \tau, \epsilon)^2 \mid x_0, c] = g_{\theta,\phi}(x_0, c)^2 + \mathrm{Var}_{\tau,\epsilon}\left(\widehat{g}_{\theta,\phi}(x_0, c; \tau, \epsilon) \mid x_0, c\right).$$

Taking expectation over $x_0 \sim \mu(\cdot \mid c)$ yields

$$\mathbb{E}_{x_0,\tau,\epsilon}[\widehat{\mathcal{J}}_{\mathrm{naive}}(\theta, \phi)] = \mathcal{J}(\theta, \phi) + \mathbb{E}_{x_0 \sim \mu(\cdot|c)}\left[\mathrm{Var}_{\tau,\epsilon}\left(\widehat{g}_{\theta,\phi}(x_0, c; \tau, \epsilon) \mid x_0, c\right)\right].$$

Since the variance term is nonnegative, the naive estimator overestimates the true objective in expectation, with strict bias whenever the variance term is positive. □

E.2 Connection to the Clean-Sample RL

Proposition 2 (ELBO calibration for rectified-flow v-prediction)**.** *Fix a condition c. Let*

$$0 = \tau_0 < \tau_1 < \cdots < \tau_T < 1$$

be a discretization of the noising time τ used in the method section. Under the rectified-flow noise scheduler,

$$x_{\tau_i} = (1 - \tau_i)x_0 + \tau_i \epsilon, \qquad \epsilon \sim \mathcal{N}(0, I),$$

and the supervised v-prediction target is $v^\star(\tau_i, x_0, \epsilon) = (1 - \tau_i)\epsilon - \tau_i x_0$. Consider the Gaussian forward chain whose marginals match this scheduler:

$$q(x_{\tau_i} \mid x_{\tau_{i-1}}) = \mathcal{N}(a_i x_{\tau_{i-1}}, s_i^2 I), \qquad a_i = \frac{1 - \tau_i}{1 - \tau_{i-1}}, \qquad s_i^2 = \tau_i^2 - a_i^2 \tau_{i-1}^2.$$

Let the reverse transition have fixed variance $p_\theta(x_{\tau_{i-1}} \mid x_{\tau_i}, c) = \mathcal{N}(\mu_{\theta,i}(x_{\tau_i}, c), \nu_i^2 I)$, where ν_i^2 is independent of θ. Parameterize the reverse mean by $\mu_{\theta,i}(x_{\tau_i}, c) = \frac{1}{a_i}\left(x_{\tau_i} - \frac{s_i^2}{\tau_i}\widehat{\epsilon}_\theta(x_{\tau_i}, \tau_i, c)\right)$, where $\widehat{\epsilon}_\theta(x_{\tau_i}, \tau_i, c) = \frac{\tau_i x_{\tau_i} + (1-\tau_i) v_\theta(x_{\tau_i}, \tau_i, c)}{(1-\tau_i)^2 + \tau_i^2}$. Let $\mathcal{E}_\theta(x_0 \mid c)$ be the per-sample ELBO of this Gaussian chain. Then there exists a function $\kappa(x_0, c)$, independent of θ, such that

$$\mathcal{L}_\theta^{\mathrm{ELBO}}(x_0; c) = -\gamma \mathcal{E}_\theta(x_0 \mid c) + \kappa_\gamma(x_0, c).$$

where $\mathcal{L}_\theta^{\mathrm{ELBO}}(x_0; c) = \gamma \sum_{i=1}^T \lambda_i \mathbb{E}_{\epsilon \sim \mathcal{N}(0,I)}\left[\|v_\theta(x_{\tau_i}, \tau_i, c) - v^\star(\tau_i, x_0, \epsilon)\|_2^2\right]$, and $\lambda_i = \frac{s_i^4 (1-\tau_i)^2}{2\nu_i^2 a_i^2 \tau_i^2 \left((1-\tau_i)^2 + \tau_i^2\right)^2}$

Proof. The chosen forward chain has the desired marginals because

$$a_i(1 - \tau_{i-1}) = 1 - \tau_i, \qquad a_i^2 \tau_{i-1}^2 + s_i^2 = \tau_i^2.$$

Hence

$$q(x_{\tau_i} \mid x_0) = \mathcal{N}((1 - \tau_i)x_0, \tau_i^2 I).$$

The exact posterior mean satisfies

$$\widetilde{\mu}_i(x_{\tau_i}, x_0) = \frac{1}{a_i}\left(x_{\tau_i} - \frac{s_i^2}{\tau_i}\epsilon\right), \qquad x_{\tau_i} = (1 - \tau_i)x_0 + \tau_i \epsilon.$$

For $i = 1$, this gives $\widetilde{\mu}_1(x_{\tau_1}, x_0) = x_0$.

The negative ELBO can be written as

$$-\mathcal{E}_\theta(x_0 \mid c) = C(x_0, c) + \sum_{i=1}^T \mathbb{E}_{q(x_{\tau_i}|x_0)}\left[\frac{1}{2\nu_i^2}\|\widetilde{\mu}_i(x_{\tau_i}, x_0) - \mu_{\theta,i}(x_{\tau_i}, c)\|_2^2\right],$$

where $C(x_0, c)$ is independent of θ. Moreover,

$$\widetilde{\mu}_i(x_{\tau_i}, x_0) - \mu_{\theta,i}(x_{\tau_i}, c) = \frac{s_i^2}{a_i \tau_i}\left(\widehat{\epsilon}_\theta(x_{\tau_i}, \tau_i, c) - \epsilon\right).$$

By the v-prediction parameterization,

$$\widehat{\epsilon}_\theta(x_{\tau_i}, \tau_i, c) - \epsilon = \frac{1-\tau_i}{(1-\tau_i)^2 + \tau_i^2} \left(v_\theta(x_{\tau_i}, \tau_i, c) - v^\star(\tau_i, x_0, \epsilon)\right).$$

Therefore,

$$\frac{1}{2\nu_i^2} \left\|\widetilde{\mu}_i(x_{\tau_i}, x_0) - \mu_{\theta,i}(x_{\tau_i}, c)\right\|_2^2 = \lambda_i \left\|v_\theta(x_{\tau_i}, \tau_i, c) - v^\star(\tau_i, x_0, \epsilon)\right\|_2^2.$$

Substituting into the ELBO decomposition gives

$$-\mathcal{E}_\theta(x_0 \mid c) = C(x_0, c) + \sum_{i=1}^{T} \lambda_i \mathbb{E}_{\epsilon\sim\mathcal{N}(0,I)} \left[\left\|v_\theta(x_{\tau_i}, \tau_i, c) - v^\star(\tau_i, x_0, \epsilon)\right\|_2^2\right].$$

Taking $\kappa = C$ proves the first claim. Multiplying by γ and absorbing $-\gamma C(x_0, c)$ into $\kappa_\gamma(x_0, c)$ proves the second claim. □

Theorem 1 (Transfer from loss-space residual matching to reward-tilted likelihood matching)**.** *Fix c. Let $\pi_\theta(x_0 \mid c)$ and $\pi_{\mathrm{ref}}(x_0 \mid c)$ be the clean-sample marginals of the current and reference models. Let $\pi^\star_\beta(x_0 \mid c) = \frac{\pi_{\mathrm{ref}}(x_0|c)\exp(\beta r(x_0,c))}{Z_\beta(c)}$ and $Z_\beta(c) = \int \pi_{\mathrm{ref}}(x_0 \mid c)\exp(\beta r(x_0, c))\, dx_0$. Assume L_θ and L_{ref} are ELBO-calibrated as in Proposition 2. Let $\mathcal{E}_\theta(x_0 \mid c) = \log \pi_\theta(x_0 \mid c) - \Delta_\theta(x_0, c)$, and $\delta_\theta(x_0, c) = \Delta_\theta(x_0, c) - \Delta_{\mathrm{ref}}(x_0, c)$. With $g_{\theta,\phi}$ defined in Eq. (4),*

$$\pi_\theta(x_0 \mid c) = \pi^\star_\beta(x_0 \mid c) \frac{\exp(g_{\theta,\phi}(x_0, c) + \delta_\theta(x_0, c))}{\mathbb{E}_{X\sim\pi^\star_\beta(\cdot|c)}[\exp(g_{\theta,\phi}(X, c) + \delta_\theta(X, c))]}.$$

Proof. By ELBO calibration,

$$\frac{1}{\gamma}\left(L_{\mathrm{ref}}(x_0; c) - L_\theta(x_0; c)\right) = \mathcal{E}_\theta(x_0 \mid c) - \mathcal{E}_{\mathrm{ref}}(x_0 \mid c).$$

Using $\mathcal{E}_\theta = \log \pi_\theta - \Delta_\theta$,

$$\frac{1}{\gamma}\left(L_{\mathrm{ref}}(x_0; c) - L_\theta(x_0; c)\right) = \log \frac{\pi_\theta(x_0 \mid c)}{\pi_{\mathrm{ref}}(x_0 \mid c)} - \delta_\theta(x_0, c).$$

Therefore,

$$g_{\theta,\phi}(x_0, c) = b_\phi(c) - \beta r(x_0, c) + \log \frac{\pi_\theta(x_0 \mid c)}{\pi_{\mathrm{ref}}(x_0 \mid c)} - \delta_\theta(x_0, c).$$

Since

$$\log \frac{\pi^\star_\beta(x_0 \mid c)}{\pi_{\mathrm{ref}}(x_0 \mid c)} = \beta r(x_0, c) - \log Z_\beta(c),$$

we obtain

$$\pi_\theta(x_0 \mid c) = \frac{Z_\beta(c)}{\exp(b_\phi(c))} \pi^\star_\beta(x_0 \mid c)\exp(g_{\theta,\phi}(x_0, c) + \delta_\theta(x_0, c)).$$

Integrating both sides over x_0 gives

$$\frac{Z_\beta(c)}{\exp(b_\phi(c))} = \left(\mathbb{E}_{X\sim\pi^\star_\beta(\cdot|c)}[\exp(g_{\theta,\phi}(X, c) + \delta_\theta(X, c))]\right)^{-1}.$$

Substitution proves the claim. □

Proposition 3 (Uniform closeness)**.** *Under the assumptions of Theorem 1, fix c. If $\|g_{\theta,\phi}(\cdot, c) + \delta_\theta(\cdot, c)\|_\infty \le \varepsilon_c$, then $D_{\mathrm{KL}}\left(\pi_\theta(\cdot \mid c) \middle\| \pi^\star_\beta(\cdot \mid c)\right) \le 2\varepsilon_c$.*

Proof. Let$h(x_0, c) = g_{\theta,\phi}(x_0, c) + \delta_\theta(x_0, c)$, by Theorem 1,

$$\log \frac{\pi_\theta(x_0 \mid c)}{\pi^\star_\beta(x_0 \mid c)} = h(x_0, c) - \log \mathbb{E}_{X\sim\pi^\star_\beta(\cdot|c)}[\exp(h(X, c))].$$

Since $\|h(\cdot, c)\|_\infty \le \varepsilon_c$,

$$-\varepsilon_c \le \log \mathbb{E}_{X\sim\pi^\star_\beta(\cdot|c)}[\exp(h(X, c))] \le \varepsilon_c.$$

Thus

$$\begin{aligned} D_{\mathrm{KL}}\left(\pi_\theta(\cdot \mid c) \middle\| \pi^\star_\beta(\cdot \mid c)\right) &= \mathbb{E}_{X\sim\pi_\theta(\cdot|c)}[h(X, c)] - \log \mathbb{E}_{X\sim\pi^\star_\beta(\cdot|c)}[\exp(h(X, c))] \\ &\le \varepsilon_c - (-\varepsilon_c) = 2\varepsilon_c. \end{aligned}$$

□

E.3 Extension to DMD-Distilled Few-Step Generators

Let $\pi_{\theta,K}(x_0 \mid c)$ be the induced DMD terminal distribution. Let $\pi_{\mathrm{ref},K}$ be the corresponding reference distribution, and define $\pi^\star_{\beta,K}(x_0 \mid c) = \frac{\pi_{\mathrm{ref},K}(x_0|c)\exp(\beta r(x_0,c))}{Z_{\beta,K}(c)}$. The same denoising network also defines an auxiliary coarse diffusion model on the K-step grid, with terminal density $p^{\mathrm{aux}}_{\theta,K}(x_0 \mid c)$. The practical DMD-FCRM residual is as defined in Appendix D.3.

For theoretical analysis, we work on a bounded domain $\Omega_c \subset \mathbb{R}^D$ which contains the support of all clean samples for a given condition c. We assume Ω_c is sufficiently regular so that the standard Poincaré inequality [6] holds (e.g., Ω_c is convex and bounded or has a Lipschitz boundary). Let $\|\cdot\|_{L^2}$ and $\|\cdot\|_{L^\infty}$ denote the usual Lebesgue norms on Ω_c.

Lemma 1 (DMD Fisher control)**.** *Fix c and let $\pi_{\theta,K}(\cdot \mid c)$ be the terminal distribution of a K-step distilled generator, and let $p^{\mathrm{aux}}_{\theta,K}(\cdot \mid c)$ be the auxiliary coarse diffusion density induced by the same denoising network on the K-step timestep grid. Assume Ω_c satisfies a Poincaré inequality: there exists $C_P(c) > 0$ such that for every smooth function f on Ω_c,*

$$\|f - \bar{f}\|_{L^2(\Omega_c)} \;\le\; C_P(c)\,\|\nabla f\|_{L^2(\Omega_c)},$$

where $\bar{f} = \frac{1}{|\Omega_c|}\int_{\Omega_c} f$. Then there exists a constant $a_{\theta,K}(c)$ (the average of $\log \frac{\pi_{\theta,K}}{p^{\mathrm{aux}}_{\theta,K}}$) such that

$$\left\|\log \frac{\pi_{\theta,K}(\cdot \mid c)}{p^{\mathrm{aux}}_{\theta,K}(\cdot \mid c)} - a_{\theta,K}(c)\right\|_{L^2(\Omega_c)} \;\le\; C_P(c)\left\|\nabla \log \pi_{\theta,K}(\cdot \mid c) - \nabla \log p^{\mathrm{aux}}_{\theta,K}(\cdot \mid c)\right\|_{L^2(\Omega_c)}.$$

In addition, the DMD loss $\mathcal{L}_{\mathrm{DMD}}(\theta; c)$ controls the score discrepancy in L^2 as

$$C_P(c)\left\|\nabla \log \pi_{\theta,K}(\cdot \mid c) - \nabla \log p^{\mathrm{aux}}_{\theta,K}(\cdot \mid c)\right\|_{L^2(\Omega_c)} \;\le\; C_{\mathrm{reg}}(c)\sqrt{\mathcal{L}_{\mathrm{DMD}}(\theta; c)},$$

then

$$\left\|\log \frac{\pi_{\theta,K}(\cdot \mid c)}{p^{\mathrm{aux}}_{\theta,K}(\cdot \mid c)} - a_{\theta,K}(c)\right\|_{L^2(\Omega_c)} \;\le\; C_{\mathrm{reg}}(c)\sqrt{\mathcal{L}_{\mathrm{DMD}}(\theta; c)}.$$

Proof. Set $f(x_0) = \log \frac{\pi_{\theta,K}(x_0|c)}{p^{\mathrm{aux}}_{\theta,K}(x_0|c)}$. Then $\nabla f = \nabla \log \pi_{\theta,K}(\cdot \mid c) - \nabla \log p^{\mathrm{aux}}_{\theta,K}(\cdot \mid c)$. Applying the Poincaré inequality to f and taking $a_{\theta,K}(c) = \bar{f}$ gives the first bound. The second bound follows directly from the DMD student score training loss. □

Theorem 2 (DMD-calibrated FCRM guarantee)**.** *Fix c. Assume the following:*

1) *FCRM residual matching: $\left\|g^K_{\theta,\phi}(\cdot, c)\right\|_{L^\infty(\Omega_c)} \;\le\; \varepsilon_{\mathrm{match}}(c)$.*
2) *ELBO calibration: The auxiliary coarse diffusion losses satisfy Proposition 2 and the posterior-gap mismatch satisfies $\left\|\delta^{\mathrm{aux}}_{\theta,K}(\cdot, c)\right\|_{L^\infty(\Omega_c)} \;\le\; \varepsilon_\delta(c)$.*
3) *DMD Loss Optimization[2]: There exist constants $a_{\theta,K}(c)$, $a_{\mathrm{ref},K}(c)$ and $C_{\mathrm{reg}}(c)$ satisfying Lemma 1.*

Then the KL divergence between the learned generator $\pi_{\theta,K}(\cdot \mid c)$ and the optimal reward-tilted distribution is bounded by

$$D_{\mathrm{KL}}\big(\pi_{\theta,K}(\cdot \mid c)\,\|\,\pi^\star_{\beta,K}(\cdot \mid c)\big) \;\le\; 2\Bigg[\varepsilon_{\mathrm{match}}(c) \;+\; C_{\mathrm{reg}}(c)\Big(\sqrt{\mathcal{L}_{\mathrm{DMD}}(\theta; c)} + \sqrt{\mathcal{L}_{\mathrm{DMD}}(\mathrm{ref}; c)}\Big) \;+\; \varepsilon_\delta(c)\Bigg].$$

[2]We need an L^∞ control on the log-ratio, which can be obtained by a standard Sobolev embedding [1] (e.g., the L^2 bound on the gradient implies an L^∞ bound on $f - \bar{f}$). In practice the DMD loss is trained to make the score difference pointwise small, so we directly assume the stronger L^∞ estimate.

Proof. From the ELBO calibration (Proposition 2) and insert it into the definition of $g^K_{\theta,\phi}$. After rearranging we obtain

$$\begin{aligned} g^K_{\theta,\phi}(x_0, c) = \widetilde{b}_\phi(c) - \beta r(x_0, c) + \log \frac{\pi_{\theta,K}(x_0 \mid c)}{\pi_{\mathrm{ref},K}(x_0 \mid c)} \\ - \Big(\log \frac{\pi_{\theta,K}(x_0 \mid c)}{p^{\mathrm{aux}}_{\theta,K}(x_0 \mid c)} - a_{\theta,K}(c)\Big) + \Big(\log \frac{\pi_{\mathrm{ref},K}(x_0 \mid c)}{p^{\mathrm{aux}}_{\mathrm{ref},K}(x_0 \mid c)} - a_{\mathrm{ref},K}(c)\Big) \\ - \delta^{\mathrm{aux}}_{\theta,K}(x_0, c), \end{aligned}$$

where $\widetilde{b}_\phi(c) = b_\phi(c) - a_{\theta,K}(c) + a_{\mathrm{ref},K}(c)$. Define $h(x_0, c) = \widetilde{b}_\phi(c) - \beta r(x_0, c) + \log \frac{\pi_{\theta,K}(x_0|c)}{\pi_{\mathrm{ref},K}(x_0|c)}$. Then from the above identity,

$$h(x_0, c) = g^K_{\theta,\phi}(x_0, c) + \Big(\log \frac{\pi_{\theta,K}}{p^{\mathrm{aux}}_{\theta,K}} - a_{\theta,K}\Big) - \Big(\log \frac{\pi_{\mathrm{ref},K}}{p^{\mathrm{aux}}_{\mathrm{ref},K}} - a_{\mathrm{ref},K}\Big) + \delta^{\mathrm{aux}}_{\theta,K}.$$

Taking L^∞ norms and applying the three assumptions yields

$$\|h(\cdot, c)\|_{L^\infty(\Omega_c)} \;\le\; \varepsilon_{\mathrm{match}}(c) + C_{\mathrm{reg}}(c)\big(\sqrt{\mathcal{L}_{\mathrm{DMD}}(\theta; c)} + \sqrt{\mathcal{L}_{\mathrm{DMD}}(\mathrm{ref}; c)}\big) + \varepsilon_\delta(c) \;=:\; B(c).$$

Now note that $\pi^\star_{\beta,K}(x_0 \mid c) = \frac{\pi_{\mathrm{ref},K}(x_0|c)\exp(\beta r(x_0,c))}{Z_{\beta,K}(c)}$. Consequently, exponentiating and normalizing over x_0 gives

$$\pi_{\theta,K}(x_0 \mid c) = \pi^\star_{\beta,K}(x_0 \mid c)\, \frac{\exp\big(h(x_0, c)\big)}{\mathbb{E}_{X\sim\pi^\star_{\beta,K}(\cdot|c)}[\exp(h(X, c))]}.$$

Because $\|h\|_{L^\infty} \le B(c)$, following the proof of Proposition 3, the KL divergence is

$$\begin{aligned} D_{\mathrm{KL}}\big(\pi_{\theta,K} \| \pi^\star_{\beta,K}\big) &= \mathbb{E}_{X\sim\pi_{\theta,K}}[h(X, c)] - \log \mathbb{E}_{X\sim\pi^\star_{\beta,K}}[\exp(h(X, c))] \\ &\le B(c) - (-B(c)) = 2B(c), \end{aligned}$$

which completes the proof. □

F Examples

F.1 Prompts

Full Prompt for Fig. 3

<Row 1> A person with a worried and puzzled expression, standing in a dimly lit room. They have a furrowed brow, slightly parted lips, and their eyes are wide with concern. Their posture is tense, with one hand pressed against their forehead. The background features shadowy walls and a single desk with scattered papers. The lighting casts dramatic shadows, highlighting the person's emotions. Medium close-up shot focusing on the person's face and upper body. </Row 1>
<Row 2> A young woman named Lily, with curly brown hair and green eyes, stands in a dimly lit room filled with ancient books and mystical artifacts. She holds a well-worn book tightly, her face illuminated by a soft, warm glow from a nearby candle. Lily wears a flowing, emerald-green robe with intricate silver embroidery, symbolizing her connection to the magical world. She takes a deep breath, her eyes narrowing in concentration as she begins to chant softly, her voice echoing in the room. The air around her swirls with a faint, shimmering mist, indicating the activation of magical energies. The scene is captured in a medium close-up, focusing on Lily's determined expression and the magical aura surrounding her.</Row 2>
<Row 3> Create a scene with a sense of emptiness and solitude. An old, abandoned house at dusk, surrounded by overgrown grass and tall weeds. The house has a dilapidated wooden porch with peeling paint and broken railings. Dark clouds fill the sky, casting long shadows across the yard. The windows are boarded up, adding to the eerie atmosphere. The scene should convey a feeling of abandonment and desolation. Wide shot, static view.</Row 3>

Full Prompt for Fig. 4

<Row 1> A serene and holy scene featuring a young brother and sister standing beside Jesus Christ. The brother, with medium-length brown hair and a gentle smile, is dressed in a traditional tunic. His sister, with long blonde braids and a joyful expression, wears a flowing white dress. They both have their hands folded in prayer, looking up at Jesus with reverence and love. Jesus, depicted with long hair and a beard, stands tall and serene, offering a comforting gaze towards the siblings. The background showcases a tranquil garden setting with lush greenery and blooming flowers, bathed in soft, golden sunlight. Medium shot, static scene focusing on the three figures.</Row 1>
<Row 2> A colossal inflatable Soviet robot from the 1950s era attacks New York City, its metallic surface gleaming under the dim city lights. The robot is adorned with Soviet-era insignias and emits vibrant red and blue lasers from its eyes and arms. It stands towering over iconic landmarks such as the Chrysler Building and the Empire State Building, causing chaos in the streets below. The scene is filled with vintage cars, bustling crowds in period attire, and a sense of impending doom. The camera captures the robot's massive form in a wide shot, emphasizing the scale and intensity of the attack.</Row 2>

Full Prompt for Fig. 5

<Row 1> A serene group of four friends, two males and two females, all in their early twenties, walking leisurely along a sandy beach at sunset. They are wearing casual summer attire, with one male carrying a backpack and the other holding a surfboard. The female friends have sun-kissed skin and are laughing and chatting animatedly. The sky is painted with soft hues of orange and pink, and the waves gently lap against the shore. The scene captures their joyful camaraderie and the peaceful ambiance of the beach. Medium shot showcasing the group moving together towards the camera. </Row 1>
<Row 2> A vibrant and colorful scene showcasing fresh vegetables in a kitchen setting. The video starts with a close-up of a crisp green lettuce leaf, then smoothly transitions to a medium shot of a large cabbage with tightly packed leaves. Next, it pans to a close-up of a bright orange carrot with its greens still attached. Each vegetable is displayed in high detail, emphasizing their textures and colors. The camera remains static during each shot, focusing solely on the individual vegetables. </Row 2>

Full Prompt for Fig. 6

<Row 1> A young woman in black leggings and a warm, cozy coat walks down a snowy street. She has fair skin, blonde hair tied in a ponytail, and wears a determined yet calm expression. Snowflakes gently fall around her as she strides confidently, leaving clear footprints behind. The street is quiet, with snow-covered buildings and trees lining both sides. The background shows a peaceful winter scene with occasional streetlights casting a soft glow. The camera follows her from a medium distance, maintaining a steady shot as she continues walking. </Row 1>
<Row 2> A realistic depiction of a powerful tornado swirling through a bustling city at dusk. The tornado is vividly detailed, with swirling red and blue clouds creating a dramatic and intense atmosphere. Debris and vehicles are being lifted and scattered by the fierce winds. The cityscape includes towering skyscrapers and smaller buildings, all affected by the storm's destructive force. The scene is captured from a mid-range aerial perspective, emphasizing the scale and power of the tornado. </Row 2>

Full Prompt for Fig. 7

<Top-left> A one-minute long cinematic video featuring a young woman walking alone on a rainy night street. The girl is dressed in a black waterproof jacket and jeans, wearing a dark hat and carrying an umbrella. Her face is illuminated by the occasional street lamp, casting dramatic shadows. The rain falls steadily, creating reflections and puddles on the wet pavement. The background showcases dimly lit buildings and neon signs. The video captures her determined stride as she navigates through the busy urban environment. Shot in a mix of close-ups and medium shots, emphasizing the atmospheric feel of a rainy city night.</Top-left>
<Top-right> A serene group of four friends, two males and two females, all in their early twenties, walking leisurely along a sandy beach at sunset. They are wearing casual summer attire, with one male carrying a backpack and the other holding a surfboard. The female friends have sun-kissed skin and are laughing and chatting animatedly. The sky is painted with soft hues of orange and pink, and the waves gently lap against the shore. The scene captures their joyful camaraderie and the peaceful ambiance of the beach. Medium shot showcasing the group moving together towards the camera.</Top-right>

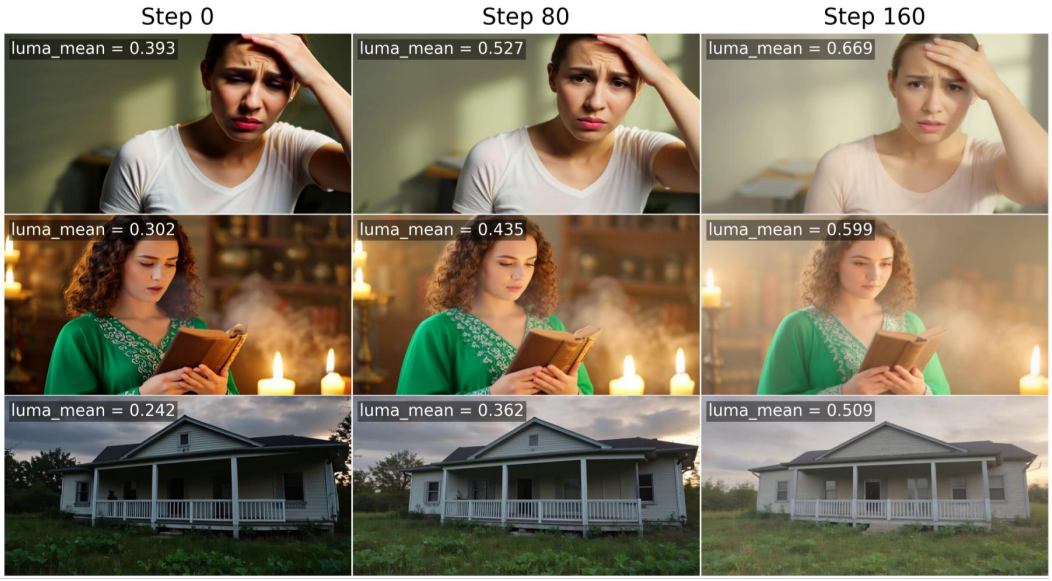

Figure 3: Fully offline FCRM training with a lightness reward. We display the outputs for four different prompts at training steps 0, 80, and 160, with the corresponding mean frame luminance (luma_mean) score overlaid on the top left of each frame.

<Bottom-left> A high-speed Phantom camera captures a detailed slow-motion sequence of a water droplet splashing into a pool, revealing intricate patterns of ripples and droplets. The droplet is perfectly round as it hits the surface, creating a mesmerizing display of fluid dynamics. The camera focuses closely on the splash, showcasing the fine details of the water interaction. The scene is set against a plain, dark background to highlight the splashing action. Close-up, static shot.</Bottom-left>
<Bottom-right> Close-up underwater perspective of fish swimming in a tranquil pond. The camera focuses on the graceful movements of the fish as they glide through the water, their scales shimmering in the soft sunlight filtering through the surface. Schools of small fish dart around gracefully, while larger fish occasionally pass by, creating gentle ripples. The background features blurred aquatic plants and sunlight beams dancing through the water. The scene is serene and full of life, with natural motion capturing the fluidity of the fish's.</Bottom-right>

F.2 Qualitative Comparison

Figure 4: Qualitative comparison of generated video frames using our proposed FCRM, AWM, and DiffusionNFT. The figure displays the initial, middle, and last frames of videos generated from two distinct text prompts. The top two rows show a serene scene featuring a brother and sister standing beside Jesus Christ, while the bottom two rows depict a colossal inflatable Soviet robot attacking New York City. See full prompts in Appendix F.1.

Figure 5: Qualitative comparison of generated video frames using our proposed FCRM, AWM, and DiffusionNFT. The figure displays the initial, middle, and last frames of videos generated from two distinct text prompts. The top two rows show a group of four friends walking leisurely along a sandy beach at sunset, while the bottom two rows display a vibrant close-up of fresh vegetables, including cabbage and a carrot, in a kitchen setting. See full prompts in Appendix F.1.

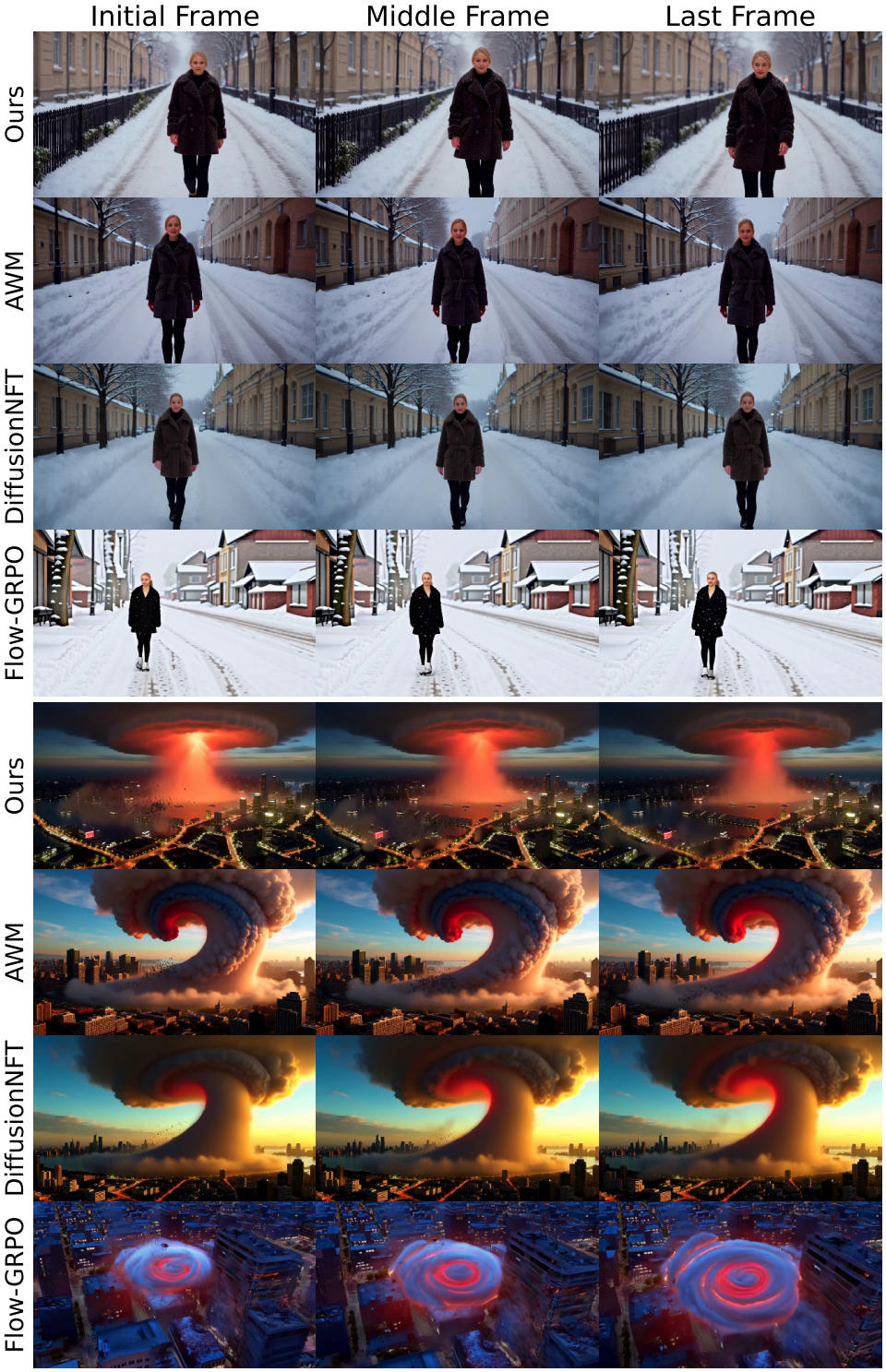

Figure 6: Qualitative comparison of generated video frames using our proposed FCRM, AWM, and DiffusionNFT. The figure displays the initial, middle, and last frames of videos generated from two distinct text prompts. The top two rows feature a young woman in a black coat walking down a snowy street, while the bottom two rows depict a powerful tornado swirling through a bustling city at dusk. See full prompts in Appendix F.1.

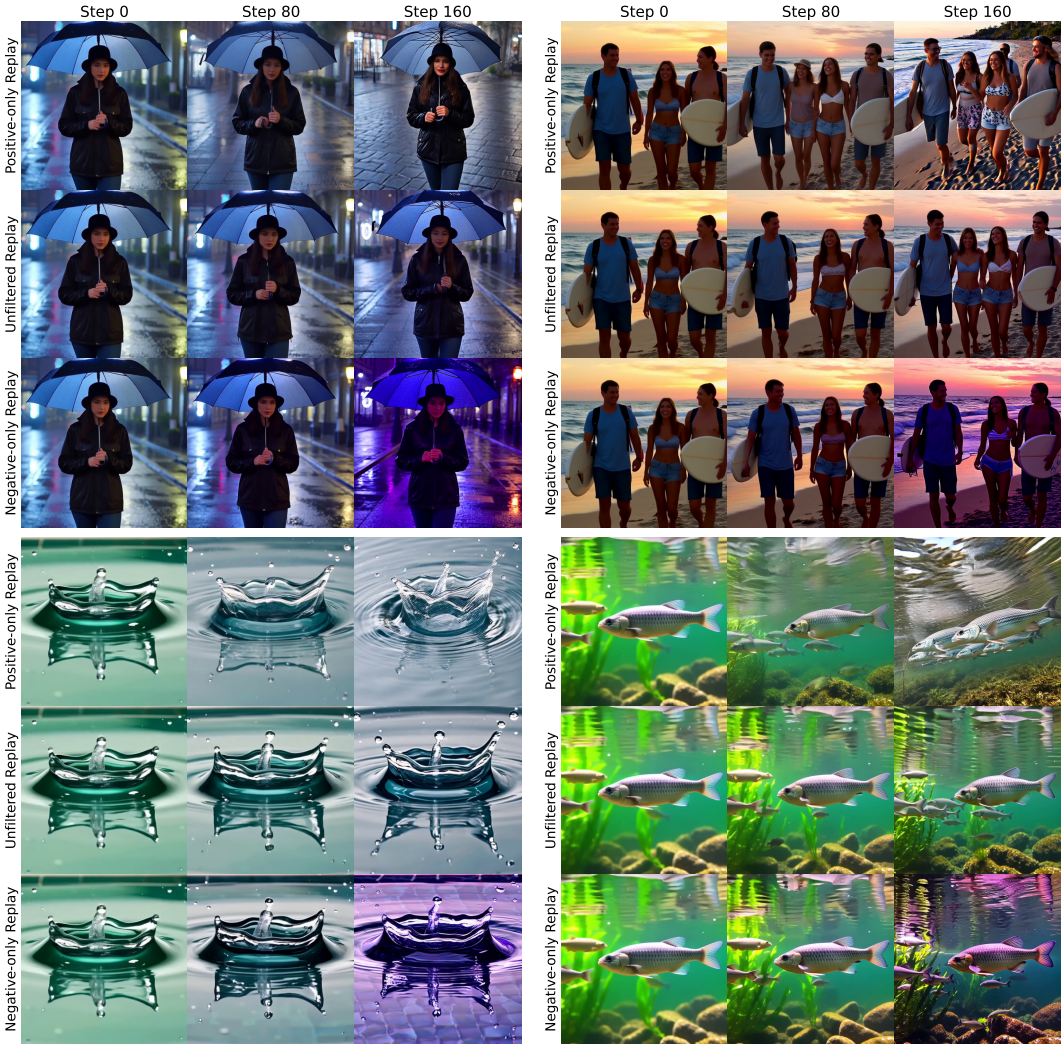

Figure 7: Qualitative comparison of generated video frames across different replay-buffer filtering strategies. The panels display sample frames generated at training steps 0, 80, and 160 using Positive-only Replay (top row), Unfiltered Replay (middle row), and Negative-only Replay (bottom row). Full prompts can be found in Appendix F.1.