Diffusion-DRF: Free, Rich, and Differentiable Reward for Video Diffusion Fine-Tuning

Yifan Wang[1,2,*] **Yanyu Li**[2] **Gordon Guocheng Qian**[2,†] **Sergey Tulyakov**[2]
Yun Fu[1] **Anil Kag**[2]
[1]Northeastern University [2]Snap Inc.

Project Page Code



Figure 1. **Text-to-video fine-tuning with Diffusion-DRF.** Our method (bottom row) enhances both text-video alignment (left) and physical fidelity (right), enabling more reliable generation under challenging prompts.

Abstract

Video diffusion alignment has been heavily reliant on scalar rewards. These rewards are typically derived from learned reward models in human preference datasets, requiring additional training and extensive collection. Moreover, scalar rewards provide coarse, global supervision, offering limited prompt-generation mismatch credit assignment and making models prone to reward exploitation and unstable optimization. We propose **Diffusion-DRF** (Differentiable Reward Flow), a free, rich, and differentiable reward framework for video diffusion fine-tuning, where "free" means free of reward-model training and of preference-data collection. Diffusion-DRF employs a frozen, off-the-shelf Vision-Language Model (VLM) as the critic, eliminating the need for reward model training. Instead of relying on a single scalar reward, it decomposes each user prompt into multi-dimensional questions with freeform dense VQA explanation queries, yielding information-rich feedback. Under teacher forcing, every answer-token loss conditions on the video, question, and a clean target prefix rather than prior model predictions; this non-compounding token supervision prevents answer errors from propagating into subsequent training contexts. By direct differentiable optimization over this rich feedback, Diffusion-DRF achieves stable reward-based tuning without preference datasets collection. Diffusion-DRF achieves significant gains both quantitatively and qualitatively, reaching 55.38 overall on VBench-2.0, 2.39 points above the base model and 1.83 points above the strongest reward-tuning baseline.

*Work done during an internship at Snap Inc. † Corresponding author.

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

1 Introduction

Recent advances in diffusion-based text-to-video generation [52, 20, 42, 60, 21, 28] have markedly improved fidelity, temporal coherence, and prompt adherence. Beyond architecture and scaling, a second wave of progress has come from post-training, inspired by alignment practices in LLMs [34, 31, 54, 38] and text-to-image diffusion [40, 16, 12, 8]. The core motivation is to use preference-driven objectives to steer pretrained generators toward human-preferred behaviors that maximum-likelihood training does not capture well. Accordingly, a growing set of post-training methods, such as preference optimization (e.g., DPO-style objectives [47, 24, 48]), reinforcement learning with human or AI feedback [49, 23, 56], and other reward-driven refinements [33, 55], have received increasing attention to align model outputs with human judgments. Despite these gains, most pipelines still rely on *hand-labeled, scalar* preference signals, either from separate reward models or large-scale DPO-style pairwise annotations. Such supervision is easily exploited due to its limited scalar supervision and label-intensive, often resulting in reward hacking, training instability, or policy collapse.



Figure 2: **Gradient Analysis.** In contrast to the latest scalar-based VideoAlign [24] and our Vanilla-DRF baseline built on a simple yes/no text-video alignment VLM feedback, our Diffusion-DRF leverages free and rich structured feedback to empirically yield more semantically coherent and spatially concentrated gradients, with heatmaps that better align with regions where the generated video appears to violate the input prompt. We present more visualizations in Section C.1.

Effective supervision should provide *token-level prompt-generation mismatch credit assignment, indicating where and when generated content mismatches the input prompt*. Such structured feedback, rather than a single scalar signal, enables more stable gradient propagation and mitigates reward hacking or training collapse. However, prior reward tuning approaches [23, 33, 55, 24] typically produce a single scalar score for an entire video. This global supervision fails to distinguish localized spatial or temporal text-video inconsistencies, allowing reinforcement learning updates to under-penalize misaligned regions while over-rewarding globally correlated artifacts. This imbalance often leads to reward exploitation and suboptimal optimization dynamics. To better understand the limitation, we conduct a gradient-level analysis of the recent scalar-reward model VideoAlign [24]. Shown in Figure 2, their gradients are spatially diffuse and frequently misplaced, failing to provide precise mismatch credit assignment.

Interestingly, we observe that a pretrained Vision-Language Model (VLM) [1], owing to its strong generalization capability, already serves as a powerful visual quality assessor. Even a simple scalar yes or no alignment signal derived from a frozen VLM forms a strong differentiable reward baseline, which we denote as **Vanilla-DRF**. Despite its simplicity, Vanilla-DRF empirically yields more focused token-level gradients, which in our visualizations better indicate where video content deviates from the prompt. Empirically, this basic VLM-based reward already surpasses the latest prior method, VideoAlign, by a significant margin (Figure 2). However, Vanilla-DRF remains inherently limited by its scalar nature, which can lead to noisy supervision and susceptibility to reward exploitation.

To address these limitations, we propose **Diffusion-DRF, a free, rich, and Differentiable Reward Flow (DRF) framework** that leverages a frozen, off-the-shelf VLM to fine-tune video diffusion models. We design an automatic prompting pipeline that elicits structured multi-dimensional feedback. Specifically, each prompt is decomposed into multiple targeted questions assessing text-video alignment, visual quality, and physical plausibility, producing binary signals across distinct alignment dimensions, complemented by freeform VQA-style explanations for dense contextual grounding. Together, these responses form *fine-grained, multi-dimensional supervision that is substantially*

denser and less prone to exploitation than conventional scalar rewards. Crucially, our teacher-forced next-token objective conditions every answer token on the generated video, its question, and a clean reference-answer prefix rather than previously predicted tokens. We call this property *non-compounding token supervision*: an erroneous token prediction cannot enter the context used to supervise subsequent tokens, avoiding error propagation along the answer sequence during training. This formulation empirically provides more precise temporally localized learning signals (Figure 2 row 4) that improve training stability and mitigate reward hacking (see Sec. 4.3), while eliminating the need for additional reward model training or large-scale preference annotation, resulting in an extensible alignment pipeline.

Our contributions are summarized as follows:

- We propose **Diffusion-DRF**, a free, rich, and differentiable reward framework for video diffusion fine-tuning. Diffusion-DRF employs a frozen, off-the-shelf VLM as a training-free critic, eliminating the need for reward model training or preference dataset collection.
- We design an automatic annotation pipeline that decomposes each prompt into multiple targeted questions with binary responses, complemented by freeform VQA explanations. The teacher-forced objective provides non-compounding token supervision, as every token is trained with a clean reference-answer prefix rather than preceding model predictions. This produces multi-dimensional, information-rich feedback that yields temporally and token-aware gradients, substantially improving stability and reducing reward hacking.
- Extensive experiments demonstrate that Diffusion-DRF achieves stable reward-based tuning under both single-prompt and large-scale settings, consistently improving generation quality and reaching 55.38 overall on VBench-2.0, 2.39 points above the base model and 1.83 points above the strongest reward-tuning baseline, VideoAlign.

2 Related Works

Diffusion Model Post-training. Recent progress in LLM post-training [30, 2, 35] has been adapted to visual generation to further improve output quality. Existing approaches can generally be divided into two paradigms: reinforcement learning–based optimization and differentiable reward optimization. RL-based methods optimize generators using preference supervision through policy-style updates. Among them, Direct Preference Optimization (DPO) [35] and Group Relative Policy Optimization (GRPO) [38] have become popular choices, as they optimize preference signals through policy-style updates. These methods have been widely explored in both image [41, 50, 61, 17, 58, 23] and video generation [47, 6, 24, 48, 59]. In parallel, differentiable-based approaches update generators using gradients derived from a reward model, enabling end-to-end optimization through the generation process [33, 55, 44, 48, 11]. Despite notable perceptual improvements, existing approaches typically rely on reward models trained from human preference data, which are annotation-intensive, narrow in coverage, and inherently biased. Moreover, these rewards are usually scalar, providing only sparse and coarse supervision. As a consequence, optimization can lead to reward hacking and unstable updates. These limitations have motivated follow-up efforts toward richer and more fine-grained feedback [22], but such approaches require even more extensive data collection.

Learning from AI Feedback Recent advancements have demonstrated that Vision–Language Models (VLMs) can transcend simple image-text matching to perform complex tasks such as aesthetic assessment, fine-grained grounding, and compositional reasoning. Thus, VLMs have functioned as automated oracles capable of judging various dimensions of visual quality which are widely used in different T2I/T2V tasks such as data selection [4, 5, 29, 39, 45] and evaluation [15, 57, 10, 13]. In particular, [3] uses VLM as a zero-shot reward model to quantify prompt–image alignment within a reinforcement learning framework. [9] leverages Gemini-generated preference data to perform DPO fine-tuning. [27] uses VLM as a differentiable scalar reward to improve multimodal control. Despite these advances, directly optimizing video diffusion models with differentiable VLM feedback remains non-trivial and largely unexplored.

3 Method

Given a text prompt c, a video diffusion model $v_\theta(\mathbf{z}_t \mid c)$ generates a video latent $\mathbf{z}_0$ through iterative denoising from Gaussian noise $\mathbf{z}_T$ in latent space, where T denotes the total number of diffusion

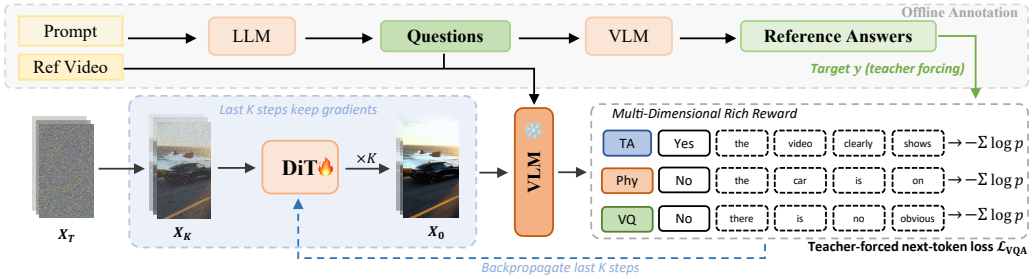


Figure 3: **Pipeline of Diffusion-DRF.** Given a reference video and its caption as the prompt, our Diffusion-DRF framework queries a frozen VLM with structured multi-dimensional questions spanning text-video alignment (TA), physical fidelity (Phy), and visual quality (VQ). For each dimension, the VLM produces a Yes/No binary judgment together with a dense free-form explanation. The logit difference through next-token prediction defines a differentiable reward for each query, and the aggregated multi-dimensional rewards are backpropagated through the VAE decoder and the final K denoising steps to update the latent diffusion model (e.g., DiT [32]).

steps. Our goal is to fine-tune the pretrained model v_θ using feedback derived from a reward function $f(\mathbf{v}, c)$, where $\mathbf{v}$ denotes the predicted clean video in pixel space decoded from $\mathbf{z}_0$.

Diffusion-DRF leverages a frozen, off-the-shelf Vision-Language Model (VLM) as a training-free critic, without learning a separate reward model or collecting preference pairs. We first introduce a minimal formulation, **Vanilla-DRF**, with a single binary reward in Section 3.1. We then extend it to our proposed structured multi-dimensional rich reward formulation in Section 3.2, and finally introduce our automatic annotation pipeline in Section 3.3.

3.1 Vanilla-DRF: Single Binary Differentiable Reward

Vanilla-DRF is a minimal formulation of Diffusion-DRF, which also provides free and differentiable VLM feedback. The difference is that Vanilla-DRF relies on a simple single binary reward.

Single Binary Reward. As a minimal formulation, we query the VLM with a single binary question: *"Does this video match the prompt?"* The VLM produces logits y over `Yes` and `No`.

Logit-level Reward. We define a differentiable reward using the log-probability:

$$R_{\text{vanilla}}(\mathbf{v}, c) = \log p(y \mid \mathbf{v}, c), \tag{1}$$

Truncated Backpropagation Through Diffusion. Let $\mathbf{z}_T \to \cdots \to \mathbf{z}_0$ denote the full denoising trajectory in training. We truncate gradients at step $T - K$ by applying a stop-gradient operator to $\mathbf{z}_{T-K}$, and backpropagate only through the final K updates:

$$\nabla_\theta R_{\text{vanilla}} = \frac{\partial R}{\partial \mathbf{v}} \frac{\partial \mathbf{v}}{\partial \mathbf{z}_0} \sum_{k=1}^{K} \frac{\partial \mathbf{z}_{k-1}}{\partial \theta}, \qquad \mathbf{z}_K \leftarrow \text{sg}(\mathbf{z}_K), \tag{2}$$

Here $\text{sg}(\cdot)$ denotes the stop-gradient operator, which treats $\mathbf{z}_{T-K}$ as constant with respect to θ. As a result, gradients flow only through the last K denoising steps for optimized memory efficiency.

3.2 Diffusion-DRF: Multi-Dimensional Rich Differentiable Reward

The single binary judgment in Vanilla-DRF compresses entity presence, motion correctness, and interaction consistency into one sparse scalar that entangles multiple failure modes: the model can raise the overall `Yes` score by exploiting superficial cues while receiving little guidance on which component of the generation should be corrected, often leading to unstable or misaligned updates (Section C.1). To obtain structured and harder-to-exploit supervision, we extend the formulation to multi-dimensional rich feedback formed by a set of targeted binary queries derived from explicit semantic and physical decompositions with their corresponding freeform dense explanations.

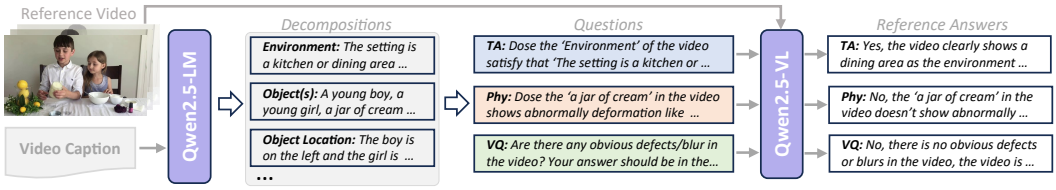


Figure 4: **Automatic Annotation Pipeline.** Instead of relying on vague global questions or human preference pairs, we construct structured queries directly from captioned videos. Given a video caption as the prompt, we extract atomic key points and formulate targeted questions across three domains: text-video alignment (TA), physical fidelity (Phy), and visual quality (VQ). Each question is posed in a minimal and unambiguous Yes/No format with a constrained free-form explanation, encouraging precise and localized judgments from the frozen VLM. For Phy and VQ, the same queries are additionally applied to caption-matched real videos to obtain reference answers that anchor physical and perceptual assessment. Detailed prompts and domain facets are provided in I.

Structured Query Set. Given a prompt c, we automatically construct a set of sub-questions for distinct dimension, $\mathcal{Q}(c) = \{q_i(c)\}_{i=1}^{N}$, where each q_i probes a specific and localized aspect of the generation. The query set is organized along three complementary dimensions in our implementation:

- *Text-Video Alignment (TA):* semantic entities, attributes, actions, and interactions explicitly described in c;
- *Physical Fidelity (Phy):* motion realism and physical plausibility;
- *Visual Quality (VQ):* perceptual fidelity including sharpness, artifacts, texture preservation, and exposure stability.

Each query q_i is instantiated as a minimal Yes/No question followed by an extra query of *free-form explanation*. The VLM is prompted to first produce a binary decision token and then generate a dense textual rationale describing where and why the video succeeds or fails for that specific dimension. Because the rationale follows the decision, we treat it as *rationale supervision* rather than chain-of-thought reasoning: its tokens provide additional, video-dependent likelihood terms, and placing the explanation before the answer performs equally well (Section E.4).

Reference-Guided Queries. For physical fidelity (Phy) and visual quality (VQ), we adopt a reference-guided evaluation mechanism. Given a caption-matched real video $\mathbf{v}^{\text{ref}}$, the VLM evaluates the generated video $\mathbf{v}$ contrastively, $f(\mathbf{v}, \mathbf{v}^{\text{ref}}, q_i)$, where the reference clip provides a physical and perceptual anchor. This stabilizes assessment of motion realism, interaction plausibility, and appearance statistics by grounding the judgment in realistic dynamics and textures.

Free, Rich, and Differentiable Rewards. For each query q_i, the VLM generates a sequence offline, $y_i = \left(y_i^0, y_i^1, \ldots, y_i^{(L)}\right)$, where the first token $y_i^0 \in \{\texttt{Yes}, \texttt{No}\}$ represents the binary judgment, and the remaining tokens form a dense free-form explanation, where $L = 512$ represents the maximum number of response tokens. This constrained format enforces localized judgments while preserving rich diagnostic information.

Diffusion-DRF Training Objective is defined as the mean next-token negative log-likelihood over all valid target tokens:

$$\mathcal{L}_{\text{VQA}} = -\frac{1}{\sum_{i,t} m_{i,t}} \sum_{i=1}^{N} \sum_{t=1}^{L_i} m_{i,t} \, \log p_\phi\left(y_i^{(t)} \mid \mathbf{c}_i, y_i^{(<t)}\right), \tag{3}$$

where the conditioning context $\mathbf{c}_i$ is defined as

$$\mathbf{c}_i = \begin{cases} (\mathbf{v}, q_i), & q_i \in \text{TA}, \\ (\mathbf{v}, \mathbf{v}^{\text{ref}}, q_i), & q_i \in \{\text{Phy}, \text{VQ}\}, \end{cases}$$

p_ϕ denotes the frozen VLM next-token distribution, and $m_{i,t} \in \{0, 1\}$ masks padding tokens. Per-query and per-dimension-balanced normalizations perform equally well (Section E.6).

Non-Compounding Token Supervision. The prefix $y_i^{(<t)}$ in the objective is the fixed reference-answer prefix supplied under teacher forcing, not a sequence sampled from the VLM. Consequently,

every token loss is conditioned on the generated video, its structured question (and the reference video for Phy and VQ), and a clean target prefix. An erroneous prediction at an earlier position therefore cannot enter the conditioning context of later positions or compound along the answer sequence. We refer to this training-time property as *non-compounding token supervision*. It removes autoregressive error propagation within the supervised answer sequence while retaining the contextual dependencies encoded by its reference-answer prefix.

This objective aggregates token-level supervision across the TA, Phy, and VQ queries: jointly optimizing the binary decisions and their explanations yields structured, localized gradients for each semantic and physical facet, which improves credit assignment and mitigates shortcut exploitation. *Overall, Diffusion-DRF is training-free with respect to reward modeling, provides rich multi-dimensional feedback, and remains fully differentiable through the VLM next-token objective.*

3.3 Automatic Annotation Pipeline

To enable the multi-dimensional supervision described above, we require structured annotations that decompose each video data into explicit semantic and physical facets. We construct such annotations automatically through a lightweight parsing and prompting pipeline, as illustrated in Figure 4.

Shown in Figure 4 "Decompositions", given a video caption as the prompt, we first decompose it into atomic scene elements in text, including background environment, objects, humans, attributes, spatial relations, and their interactions. This decomposition identifies the explicit entities and behaviors that should appear in the video. Concretely, a text-only LLM (Qwen2.5-7B) is prompted with the instruction template in Figures 13 and 14 to list these elements, and a second instruction turns them into questions using the dimension-specific templates in Figures 15 to 17. The extracted components are then incorporated into the formulation of one structured Yes/No question for each supervision dimension shown in "Questions". Each dimension therefore produces one Yes/No judgment accompanied by a dense free-form explanation that references the decomposed components. The decomposed entities and interactions are also embedded directly into both the question wording and the expected explanations (Figure 4 "Reference Answers"), ensuring that the VLM grounds its judgment in concrete semantic and physical details.

Reference answers are generated once with Qwen2.5-VL-7B under effectively greedy decoding (Section B.4): the paired real video is answered for Phy and VQ, and only samples whose reference answers are all `Yes` are kept (Section B.2). The resulting training set contains 5,077 caption–video pairs and 24,252 queries, with a TA:Phy:VQ ratio of roughly 2.5:1:1. The three dimensions are operational supervision groups rather than a complete factorization of video quality; further groups, such as long-range temporal consistency, can be added without changing the objective. This automated annotation pipeline replaces vague global evaluation with rich, grounded, aspect-aware supervision. It requires captioned real-video pairs for reference-guided training, avoids reward model training and human preference collection, and scales naturally to large datasets. By injecting explicit semantic and physical elements into both the question and explanation, the resulting feedback becomes more informative, interpretable, and stable for diffusion fine-tuning.

4 Experiments

4.1 Experimental Setup

Training Details. We apply Diffusion-DRF to the pretrained Wan2.1-1.3B-T2V [42] with Qwen2.5-VL-7B [1] as the default VLM. We only train the DiT and freeze other components (VAE, text-encoder, and VLM). Videos are generated at 512×288 resolution with 49 frames under 25 denoise steps during training and 30 steps at inference. AdamW [26, 18] optimizer is used with a learning rate of $1e^{-5}$, 16 A100 80GB GPUs, and batch size 1 per GPU. The training is conducted on 5K high-quality caption–video pairs from OpenVid-1M [29] (see Sections B.2 and B.3): each caption prompts an on-policy sample from the generator, and its paired real video serves only as the Phy/VQ reference. Real videos are never used as reconstruction targets, and inference is reference-free. The automatic annotation pipeline is processed offline by Qwen2.5-7B.

Efficiency Optimization. Back-propagating VLM feedback through the full diffusion chain is prohibitively memory-intensive, as both the DiT denoiser and the frozen VLM are multi-billion parameter models with large activation footprints. A naive implementation quickly exceeds GPU

Table 1: **Quantitative results on VBench-2.0.** In addition to the overall score, we report representative sub-dimension metrics closely related to text-video alignment and physical fidelity. The best results are highlighted in **bold**, and the second best are underlined. For completeness, we evaluate Diffusion-DRF with three different VLM backbones: InternVL3-1B [62], Qwen2.5-VL-3B, and Qwen2.5-VL-7B [1], denoted as Diffusion-DRF (1B), (3B), and (7B), respectively. All methods are trained with Wan2.1-1.3B-T2V as base model on the same datasets for fair comparisons.

Method	*VBench-2.0*										
	Overall	Creativity	Common Sense	Controllability	Human Fidelity	Physics	Material	Dynamic Attribute	Motion Rationality	Complex Landscape	Cameras Motion
Wan2.1-1.3B-T2V [42]	52.99	53.79	55.52	26.59	80.65	48.40	36.23	37.00	37.36	17.33	20.68
Flow-DPO [24]	50.27	46.77	50.28	<u>27.76</u>	71.76	54.78	69.24	40.29	35.06	<u>18.44</u>	22.24
Flow-GRPO [23]	50.64	44.71	50.85	25.48	77.80	54.37	69.07	36.63	36.21	14.89	19.32
PickScore [19]	49.62	35.97	55.23	23.88	**81.89**	51.13	66.67	39.19	36.78	16.89	22.84
VideoAlign [24]	53.55	49.64	56.09	26.32	<u>81.43</u>	54.26	65.26	41.76	37.93	<u>18.44</u>	23.15
Vanilla-DRF (7B)	53.70	52.70	54.66	26.64	78.55	55.61	65.31	36.63	36.21	16.67	19.44
Diffusion-DRF (1B)	53.10	50.85	56.90	24.33	77.31	<u>56.21</u>	68.63	40.30	40.20	15.11	**28.40**
Diffusion-DRF (3B)	<u>53.72</u>	<u>53.93</u>	**57.53**	25.95	76.74	54.42	<u>70.00</u>	<u>42.49</u>	**41.38**	18.22	23.15
Diffusion-DRF (7B)	**55.38**	**54.58**	<u>56.96</u>	**27.98**	80.51	**56.85**	**75.82**	**42.86**	<u>40.23</u>	**21.56**	<u>24.69</u>

VRAM limits. To make differentiable reward flow practically feasible, we design a set of optimizations: *1) Efficient Video Decoding.* Instead of using the original heavy decoder, we trained an efficient H3AE VAE decoder following [46] that reproduces the decoding quality of the original decoder with a lighter backbone and employ it only in the reward path, achieving 24.3% less memory consumption in decoding; swapping it for the original decoder changes VBench-2.0 Overall by only 0.3 points (Section E.7). *2) Frame Subsampling.* To further control VLM activation growth with video length, we uniformly subsample frames before feeding them to the VLM. In practice, we sample 10 frames from the generated video and 10 reference frames from a caption-matched real video, which preserves semantic coverage while substantially reducing memory usage (see Section E.2 for ablation). *3) Denoising Aware Activation Checkpointing.* We apply gradient checkpointing across diffusion steps to trade additional computation for lower memory usage. Concretely, we retain only the input latent at each denoising step and re-compute intermediate DiT activations during the backward pass. This avoids storing the full denoising trajectory in memory, enabling stable end-to-end gradient propagation through the diffusion chain. *4) Truncated Back-propagation.* While checkpointing allows backpropagation through the entire sampling chain, we find that truncating the backward pass to the last K sampling steps (see Equation (2)) substantially improves optimization speed and overall performance [7, 14]. This yields a biased but empirically effective gradient estimator, whose performance–memory trade-off we ablate in Section E.3.

Baselines. We compare Diffusion-DRF to reinforcement learning based methods Flow-DPO [24] and Flow-GRPO [23]. For differentiable reward tuning approaches, we compare to PickScore [19] and VideoAlign [24], which only provide scalar rewards. All methods fine-tune the same Wan2.1-1.3B-T2V [42] as the base model on the same dataset for fair comparisons. We evaluate the checkpoints of different training steps and report the best result for each model; full training curves are shown in Figure 6, and three independent runs of Diffusion-DRF give 55.44 ± 0.26 (Section D.4).

Evaluation. We evaluate text-to-video performance on VBench-2.0 [57], which provides $1,013$ prompts covering advanced aspects from human actions to physical phenomena and aggregates 18 sub-dimensions into five axes: *creativity, commonsense, controllability, human fidelity, and physics*. VBench-2.0 is a multi-faceted evaluation suite rather than a single model judge: its sub-dimensions combine generalist evaluators with specialist metrics and are calibrated with human annotations. We also use the VideoGen-Eval prompt set [51], which includes 400 instruction-heavy prompts designed to stress text–video alignment with a pairwise evaluation.

4.2 Main Results

Quantitative results on VBench-2.0. As shown in Table 1, Diffusion-DRF improves the base model across nearly all dimensions with different frozen VLM as reward models. The gains are particularly pronounced on *Dynamic Attribute* and *Physics*, reflecting better adherence to complex prompts and stronger physical plausibility. Particularly, scaling the reward model from 1B to 7B yields additional improvements, consistent with the stronger feedback provided by a better VLM. Compared to state-of-the-art reinforcement learning tuning methods Flow-DPO [24] and Flow-GRPO [23], our approach consistently improves performance as indicated in "Overall" score and

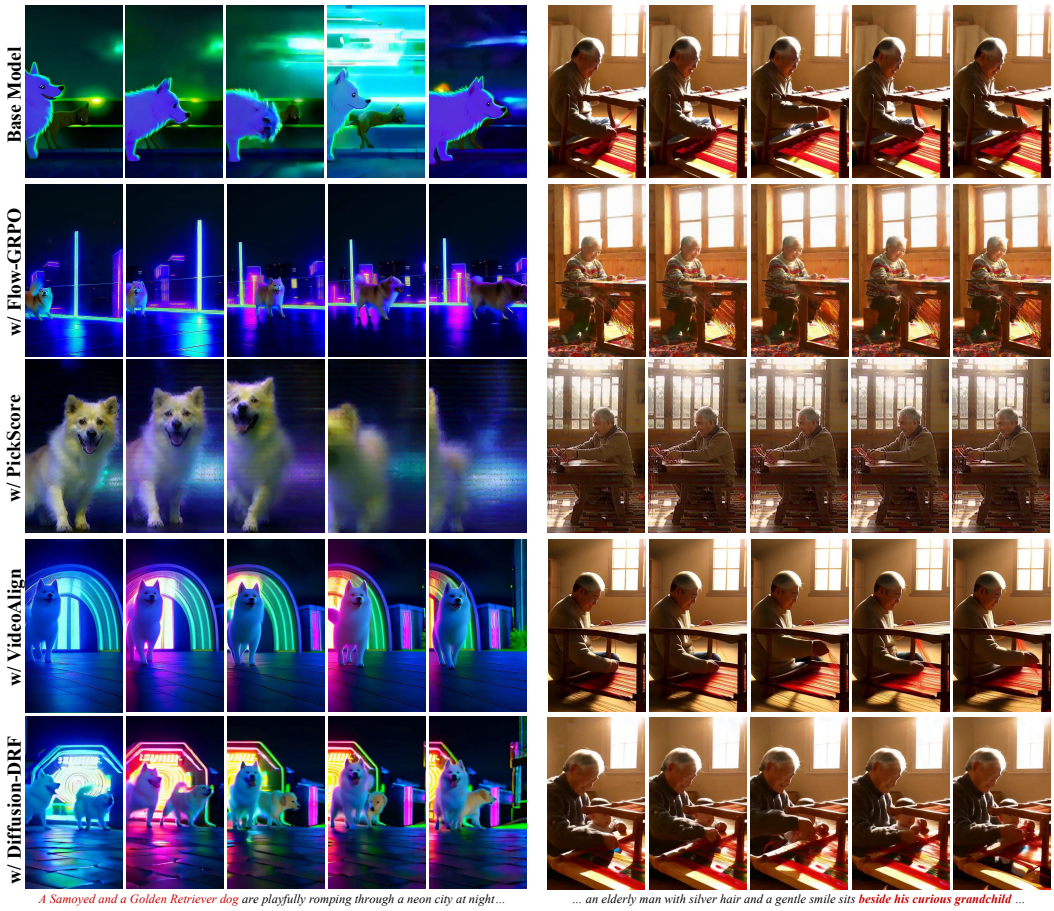


Figure 5: **Qualitative Comparison.** All videos are generated under the same configuration and random seed. Across both examples, prior reward-tuning baselines either miss key entities or relations in the prompt, or noticeably degrade video quality, while our method better preserves semantic fidelity and visual coherence. For example, competing methods fail to correctly model two different dogs under complex lighting in the left case and omit the child in the right case, whereas our approach faithfully captures both the described motion and entities. See supplementary videos for full video comparisons across methods.

achieves much higher quality in "Creativity", "Common Sense", "Materials", "Dynamic Attributes", "Motion Rationality", and "Complex Landscape", demonstrating a better prompt following.

Compared with training dedicated scalar reward models, such as PickScore [19] and VideoAlign [24], our method leverages a frozen VLM and achieves consistently stronger "Overall" performance: 55.38 versus 53.55 for VideoAlign, the strongest baseline, and 52.99 for the base model. Because Diffusion-DRF keeps improving instead of collapsing, its selected checkpoint is trained longer, so this 1.83-point gain over VideoAlign comes at about 3.2$\times$ its GPU-hours, while the gain over Flow-GRPO comes at a slightly lower cost (Table 5). In practice, these customized reward models are often susceptible to reward hacking, exhibiting biases toward particular visual styles, for example prompt-unrequested stylization (Figures 5 and 7), which ultimately harms creativity and controllability. Consistently, VBench-2.0 Diversity drops from 58.6 for the base model to 29.5 for PickScore and 52.4 for VideoAlign, whereas Diffusion-DRF raises it to 64.4. Our approach, without reward model training, delivers clear improvements on these dimensions as well as "Physics", "Creativity", and "Common Sense" metrics.

Pairwise Evaluation. To complement VBench-2.0, we perform pairwise evaluation on VideoGen-Eval between the base model and Diffusion-DRF. Table 2 reports both automatic preference scores from VideoAlign [24], a reward model trained on human-preference data, and blind human judgments on text-video alignment over 200 randomly sampled prompts, with three matched seeds per prompt and sixteen annotators (9,600 judgments; protocol in Section D.3). Under VideoAlign, Diffusion-DRF wins on 53.3% of pairs and loses on 35.7%. Human annotators prefer our method in 29.0% of cases versus 17.0% for the base model, a net preference of +12.0 points (95% CI $[+5.0, +19.0]$, Krippendorff's $\alpha = 0.72$); among decisive comparisons, humans prefer Diffusion-DRF in 63.0% of cases, close to the 59.9% under VideoAlign. Because the comparator is the base model, this study establishes improvement over the base model with an evaluator independent of our training signal, not a ranking against the other baselines. Additional pairwise comparisons against Flow-GRPO and per-metric breakdowns are provided in Section D.2.

Table 2: Pairwise evaluation of Diffusion-DRF vs. the base model, including VideoAlign-based automatic preference and human judgments.

Metrics	Win (%)	Tie (%)	Lose (%)
VideoAlign	**53.3**	11.0	35.7
Human	**29.0**	54.0	17.0

Qualitative results. Fig. 5 shows that Diffusion-DRF produces videos with stronger semantic fidelity and visual coherence than the base model, Flow-GRPO [23], VideoAlign [24], and PickScore [19], which often miss key entities or degrade visual quality.

4.3 In-depth Analysis of Diffusion-DRF

Training dynamics analysis. To examine how different reward formulations affect optimization stability, Fig. 6 tracks the VBench-2.0 'Overall' score over an extended training horizon. *Notably, Diffusion-DRF is the only method whose performance consistently remains above the base model throughout the entire training process.* In contrast, PickScore [19] deteriorates early and drops far below the base model, indicating clear reward hacking; VideoAlign [24] improves in the early phase but gradually declines afterward, again suggesting reward exploitation; and Vanilla-DRF, driven by a single scalar reward, shows the same instability. A likely explanation is that fine-tuning a foundation model into a narrow scalar reward predictor weakens its general video understanding, so it prioritizes superficial cues over the holistic behaviors we aim to improve. In contrast, Diffusion-DRF keeps the VLM frozen to preserve its general video understanding capacity and leverages multi-dimensional feedback to provide structured supervision that is harder to exploit.

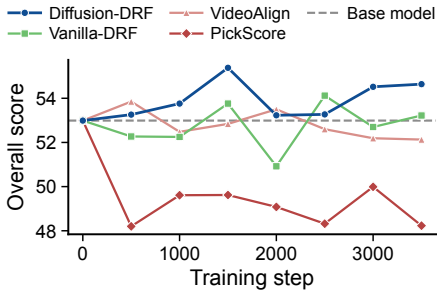


Figure 6: **Training dynamics** of the same base model optimized with different rewards.

Single-Prompt Reward Tuning. Section 4.2 demonstrates that Diffusion-DRF consistently outperforms prior reward tuning approaches. To provide a more controlled and intuitive comparison, similar to DreamBooth [37], which fine-tunes a base model on a single concept, *we introduce a single-prompt reward tuning setting*, where the model parameters are optimized using only one prompt under the guidance of a reward model for 100 steps. This controlled setting allows us to more directly examine whether a reward model can provide effective supervision. Shown in Fig. 7, when the base model fails to generate

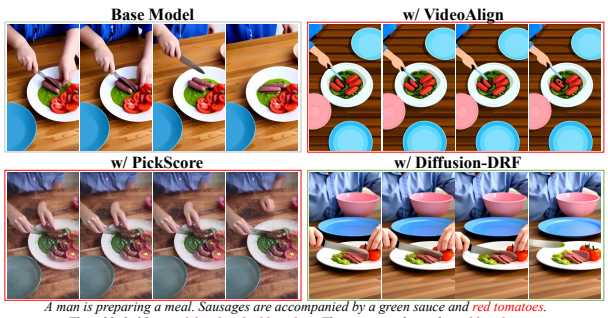


Figure 7: **Qualitative results of single-prompt reward tuning.** The base model, VideoAlign [24], and PickScore [19] all exhibit noticeable defects. In contrast, Diffusion-DRF produces the most faithful result without introducing visual distortions.

a satisfactory video, learned reward models such as PickScore [19] and VideoAlign [24] fail to deliver informative signals about where and how the generation is incorrect, leading to clear overfitting: the generated videos show significant style shift or noticeable visual distortions, where the text-video alignment remains low.

4.4 Ablation Study

Effect of multi-dimensional feedback. We study the effect of multi-dimensional feedback varying the question set used for training. All single-dimension settings (TA only, VQ only, and Phy only) underperform the baseline, suggesting that any isolated objective is easy to exploit. Once two dimensions are combined, all variants surpass the baseline, and TA+VQ is the strongest pair, indicating that explicit visual-quality supervision helps stabilize alignment-oriented optimization. Using all three dimensions yields the best result by a clear margin, improving the overall score from 52.99 to 55.38. These results support our core design choice: multi-dimensional questions provide complementary constraints that reduce reward hacking and lead to more balanced optimization. We also provide the detailed results in Section E.1.

Effect of explanation. We study the role of free-form explanations through two ablations: (1) removing the explanation and keeping only binary questions; and (2) varying the length of the reference explanations used for supervision. When only binary questions are used, the feedback becomes significantly less informative, resulting in weaker optimization and degraded performance, as shown in Table 3 "Binary question". This confirms that binary signals alone provide limited guidance for correcting fine-grained errors. The gain comes from the content of the explanations rather than from their length: a length-matched control that pairs each query with a shuffled explanation scores 52.47, below the answer-only objective (Section E.4). We further analyze the impact of explanation length. By default, we use reference explanations with an average length of 50 tokens. We then adopt Chain-of-Thought (CoT) prompting to generate detailed reference explanations (denoted as "Explanation w/ CoT" in the table). CoT-style explanations help early in training (54.85 at 2K steps) but collapse earlier than concise explanations (51.89 at 3K steps), which we attribute to verbose, weakly grounded tokens diluting the gradient; see Section E.4.

Table 3: **Ablation studies on question sets and styles.**

Question sets	*VBench-2.0*
Baseline	52.99
TA Only	52.15
VQ Only	52.45
Phy Only	51.81
TA + Phy	53.55
TA + VQ	<u>53.92</u>
VQ + Phy	53.20
TA + VQ + Phy	**55.38**
Binary question	53.30
Explanation w/ CoT	54.85

5 Conclusion

We introduced **Diffusion-DRF**, a video diffusion post-training framework that uses rich, differentiable feedback from a frozen VLM. Instead of training a separate scalar reward model, our method extracts logit-level signals for text-video alignment, physical fidelity, and visual quality, while free-form explanations provide additional corrective supervision. With efficient video decoding, frame subsampling, denoising-aware activation checkpointing, and truncated backpropagation, Diffusion-DRF makes such optimization practical for large video generators. Extensive experiments show consistent gains in prompt adherence, physical plausibility, and perceptual quality, while reducing reward hacking without requiring reward-model training or large-scale preference data.

Limitations. The Phy and VQ dimensions require a caption-matched real video for every training prompt, and the gain is contingent on matched references: without them, or with shuffled ones, the score is within run-to-run variance of the base model (Section E.5). Our critic and evaluators are both learned models; the human study provides independent evidence only against the base model. The objective scores one cached wording of each explanation, truncated back-propagation and the H3AE decoder are approximations whose effect we characterize only empirically, the critic sees 10 frames of 49-frame clips, and we test critics only up to 7B parameters. Finally, Diffusion-DRF still collapses beyond about 3K steps, and its longer stable training costs more GPU-hours than VideoAlign. We discuss these limitations and future directions in detail in Section F.

References

[1] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*, 2025.

[2] Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, Carol Chen, Catherine Olsson, Christopher Olah, Danny Hernandez, Dawn Drain, Deep Ganguli, Dustin Li, Eli Tran-Johnson, Ethan Perez, Jamie Kerr, Jared Mueller, Jeffrey Ladish, Joshua Landau, Kamal Ndousse, Kamile Lukosuite, Liane Lovitt, Michael Sellitto, Nelson Elhage, Nicholas Schiefer, Noemi Mercado, Nova DasSarma, Robert Lasenby, Robin Larson, Sam Ringer, Scott Johnston, Shauna Kravec, Sheer El Showk, Stanislav Fort, Tamera Lanham, Timothy Telleen-Lawton, Tom Conerly, Tom Henighan, Tristan Hume, Samuel R. Bowman, Zac Hatfield-Dodds, Ben Mann, Dario Amodei, Nicholas Joseph, Sam McCandlish, Tom Brown, and Jared Kaplan. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*, 2022.

[3] Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning. *arXiv preprint arXiv:2305.13301*, 2023.

[4] Lin Chen, Xilin Wei, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Bin Lin, Zhenyu Tang, Li Yuan, Yu Qiao, Dahua Lin, Feng Zhao, and Jiaqi Wang. Sharegpt4video: Improving video understanding and generation with better captions. *Advances in Neural Information Processing Systems*, 37:19472–19495, 2024.

[5] Tsai-Shien Chen, Aliaksandr Siarohin, Willi Menapace, Ekaterina Deyneka, Hsiang wei Chao, Byung Eun Jeon, Yuwei Fang, Hsin-Ying Lee, Jian Ren, Ming-Hsuan Yang, and Sergey Tulyakov. Panda-70m: Captioning 70m videos with multiple cross-modality teachers, 2024. URL `https://arxiv.org/abs/2402.19479`.

[6] Guo Cheng, Danni Yang, Ziqi Huang, Jianlou Si, Chenyang Si, and Ziwei Liu. Realdpo: Real or not real, that is the preference. *arXiv preprint arXiv:2510.14955*, 2025.

[7] Kevin Clark, Paul Vicol, Kevin Swersky, and David J Fleet. Directly fine-tuning diffusion models on differentiable rewards. *arXiv preprint arXiv:2309.17400*, 2023.

[8] 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. *NeurIPS*, 2023.

[9] Hiroki Furuta, Heiga Zen, Dale Schuurmans, Aleksandra Faust, Yutaka Matsuo, Percy Liang, and Sherry Yang. Improving dynamic object interactions in text-to-video generation with ai feedback. *arXiv preprint arXiv:2412.02617*, 2024.

[10] Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. Geneval: An object-focused framework for evaluating text-to-image alignment. *Advances in Neural Information Processing Systems*, 36:52132–52152, 2023.

[11] Xuan He, Dongfu Jiang, Ge Zhang, Max Ku, Achint Soni, Sherman Siu, Haonan Chen, Abhranil Chandra, Ziyan Jiang, Aaran Arulraj, Kai Wang, Quy Duc Do, Yuansheng Ni, Bohan Lyu, Yaswanth Narsupalli, Rongqi Fan, Zhiheng Lyu, Yuchen Lin, and Wenhu Chen. VideoScore: Building automatic metrics to simulate fine-grained human feedback for video generation. In *EMNLP*, 2024.

[12] Jiwoo Hong, Sayak Paul, Noah Lee, Kashif Rasul, James Thorne, and Jongheon Jeong. Margin-aware preference optimization for aligning diffusion models without reference. In *ICLR Workshop*, 2025.

[13] Yushi Hu, Benlin Liu, Jungo Kasai, Yizhong Wang, Mari Ostendorf, Ranjay Krishna, and Noah A Smith. Tifa: Accurate and interpretable text-to-image faithfulness evaluation with question answering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 20406–20417, 2023.

[14] Xun Huang, Zhengqi Li, Guande He, Mingyuan Zhou, and Eli Shechtman. Self forcing: Bridging the train-test gap in autoregressive video diffusion, 2025. URL https://arxiv.org/abs/2506.08009.

[15] Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, Yaohui Wang, Xinyuan Chen, Limin Wang, Dahua Lin, Yu Qiao, and Ziwei Liu. VBench: Comprehensive benchmark suite for video generative models. In *CVPR*, 2024.

[16] Shyamgopal Karthik, Huseyin Coskun, Zeynep Akata, Sergey Tulyakov, Jian Ren, and Anil Kag. Scalable ranked preference optimization for text-to-image generation. *arXiv preprint arXiv:2410.18013*, 2024.

[17] Shyamgopal Karthik, Huseyin Coskun, Zeynep Akata, Sergey Tulyakov, Jian Ren, and Anil Kag. Scalable ranked preference optimization for text-to-image generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 18399–18410, 2025.

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

[19] 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. *NeurIPS*, 2023.

[20] Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, Kathrina Wu, Qin Lin, Junkun Yuan, Yanxin Long, Aladdin Wang, Andong Wang, Changlin Li, Duojun Huang, Fang Yang, Hao Tan, Hongmei Wang, Jacob Song, Jiawang Bai, Jianbing Wu, Jinbao Xue, Joey Wang, Kai Wang, Mengyang Liu, Pengyu Li, Shuai Li, Weiyan Wang, Wenqing Yu, Xinchi Deng, Yang Li, Yi Chen, Yutao Cui, Yuanbo Peng, Zhentao Yu, Zhiyu He, Zhiyong Xu, Zixiang Zhou, Zunnan Xu, Yangyu Tao, Qinglin Lu, Songtao Liu, Dax Zhou, Hongfa Wang, Yong Yang, Di Wang, Yuhong Liu, Jie Jiang, and Caesar Zhong. HunyuanVideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024.

[21] Jiachen Li, Qian Long, Jian Zheng, Xiaofeng Gao, Robinson Piramuthu, Wenhu Chen, and William Yang Wang. T2V-Turbo-v2: Enhancing video generation model post-training through data, reward, and conditional guidance design. In *ICLR*, 2025.

[22] Youwei Liang, Junfeng He, Gang Li, Peizhao Li, Arseniy Klimovskiy, Nicholas Carolan, Jiao Sun, Jordi Pont-Tuset, Sarah Young, Feng Yang, Junjie Ke, Krishnamurthy Dj Dvijotham, Katie Collins, Yiwen Luo, Yang Li, Kai J Kohlhoff, Deepak Ramachandran, and Vidhya Navalpakkam. Rich human feedback for text-to-image generation. In *CVPR*, 2024.

[23] 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. *arXiv preprint arXiv:2505.05470*, 2025.

[24] Jie Liu, Gongye Liu, Jiajun Liang, Ziyang Yuan, Xiaokun Liu, Mingwu Zheng, Xiele Wu, Qiulin Wang, Wenyu Qin, Menghan Xia, Xintao Wang, Xiaohong Liu, Fei Yang, Pengfei Wan, Di Zhang, Kun Gai, Yujiu Yang, and Wanli Ouyang. Improving video generation with human feedback. *arXiv preprint arXiv:2501.13918*, 2025.

[25] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, and Lei Zhang. Grounding DINO: Marrying DINO with grounded pre-training for open-set object detection. In *ECCV*, 2024.

[26] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019.

[27] Grace Luo, Jonathan Granskog, Aleksander Holynski, and Trevor Darrell. Dual-process image generation. *arXiv preprint arXiv:2506.01955*, 2025.

[28] Willi Menapace, Aliaksandr Siarohin, Ivan Skorokhodov, Ekaterina Deyneka, Tsai-Shien Chen, Anil Kag, Yuwei Fang, Aleksei Stoliar, Elisa Ricci, Jian Ren, and Sergey Tulyakov. Snap video: Scaled spatiotemporal transformers for text-to-video synthesis. *CVPR*, 2024.

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

[30] Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. URL `https://proceedings.neurips.cc/paper_files/paper/2022/file/b1efde53be364a73914f58805a001731-Paper-Conference.pdf`.

[31] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In *NeurIPS*, 2022.

[32] William Peebles and Saining Xie. Scalable diffusion models with transformers. *ICCV*, 2023.

[33] Mihir Prabhudesai, Russell Mendonca, Zheyang Qin, Katerina Fragkiadaki, and Deepak Pathak. Video diffusion alignment via reward gradients. *arXiv preprint arXiv:2407.08737*, 2024.

[34] 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. *NeurIPS*, 2023.

[35] 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.

[36] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. SAM 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024.

[37] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *CVPR*, 2023.

[38] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

[39] Jiao Sun, Deqing Fu, Yushi Hu, Su Wang, Royi Rassin, Da-Cheng Juan, Dana Alon, Charles Herrmann, Sjoerd Van Steenkiste, Ranjay Krishna, et al. Dreamsync: Aligning text-to-image generation with image understanding feedback. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 5920–5945, 2025.

[40] 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 *CVPR*, 2024.

[41] 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.

[42] 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.

[43] Wenhao Wang and Yi Yang. VideoUFO: A million-scale user-focused dataset for text-to-video generation. *arXiv preprint arXiv:2503.01739*, 2025.

[44] Yibin Wang, Zhiyu Tan, Junyan Wang, Xiaomeng Yang, Cheng Jin, and Hao Li. LiFT: Leveraging human feedback for text-to-video model alignment. *arXiv preprint arXiv:2412.04814*, 2024.

[45] Xun Wu, Shaohan Huang, Guolong Wang, Jing Xiong, and Furu Wei. Boosting text-to-video generative model with mllms feedback. *Advances in Neural Information Processing Systems*, 37:139444–139469, 2024.

[46] Yushu Wu, Yanyu Li, Ivan Skorokhodov, Anil Kag, Willi Menapace, Sharath Girish, Aliaksandr Siarohin, Yanzhi Wang, and Sergey Tulyakov. H3ae: High compression, high speed, and high quality autoencoder for video diffusion models. *arXiv preprint arXiv:2504.10567*, 2025.

[47] Ziyi Wu, Anil Kag, Ivan Skorokhodov, Willi Menapace, Ashkan Mirzaei, Igor Gilitschenski, Sergey Tulyakov, and Aliaksandr Siarohin. Densedpo: Fine-grained temporal preference optimization for video diffusion models. *arXiv preprint arXiv:2506.03517*, 2025.

[48] Jiazheng Xu, Yu Huang, Jiale Cheng, Yuanming Yang, Jiajun Xu, Yuan Wang, Wenbo Duan, Shen Yang, Qunlin Jin, Shurun Li, Jiayan Teng, Zhuoyi Yang, Wendi Zheng, Xiao Liu, Ming Ding, Xiaohan Zhang, Xiaotao Gu, Shiyu Huang, Minlie Huang, Jie Tang, and Yuxiao Dong. VisionReward: Fine-grained multi-dimensional human preference learning for image and video generation. *arXiv preprint arXiv:2412.21059*, 2024.

[49] Zeyue Xue, Jie Wu, Yu Gao, Fangyuan Kong, Lingting Zhu, Mengzhao Chen, Zhiheng Liu, Wei Liu, Qiushan Guo, Weilin Huang, and Ping Luo. DanceGRPO: Unleashing grpo on visual generation. *arXiv preprint arXiv:2505.07818*, 2025.

[50] Kai Yang, Jian Tao, Jiafei Lyu, Chunjiang Ge, Jiaxin Chen, Weihan Shen, Xiaolong Zhu, and Xiu Li. Using human feedback to fine-tune diffusion models without any reward model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8941–8951, 2024.

[51] Yuhang Yang, Ke Fan, Shangkun Sun, Hongxiang Li, Ailing Zeng, FeiLin Han, Wei Zhai, Wei Liu, Yang Cao, and Zheng-Jun Zha. Videogen-eval: Agent-based system for video generation evaluation. *arXiv preprint arXiv:2503.23452*, 2025.

[52] Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, Da Yin, Yuxuan Zhang, Weihan Wang, Yean Cheng, Bin Xu, Xiaotao Gu, Yuxiao Dong, and Jie Tang. CogVideoX: Text-to-video diffusion models with an expert transformer. In *ICLR*, 2025.

[53] 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, 2025. URL `https://arxiv.org/abs/2412.07772`.

[54] Tianyu Yu, Yuan Yao, Haoye Zhang, Taiwen He, Yifeng Han, Ganqu Cui, Jinyi Hu, Zhiyuan Liu, Hai-Tao Zheng, Maosong Sun, and Tat-Seng Chua. RLHF-V: Towards trustworthy mllms via behavior alignment from fine-grained correctional human feedback. In *CVPR*, 2024.

[55] Hangjie Yuan, Shiwei Zhang, Xiang Wang, Yujie Wei, Tao Feng, Yining Pan, Yingya Zhang, Ziwei Liu, Samuel Albanie, and Dong Ni. InstructVideo: Instructing video diffusion models with human feedback. In *CVPR*, 2024.

[56] Daoan Zhang, Guangchen Lan, Dong-Jun Han, Wenlin Yao, Xiaoman Pan, Hongming Zhang, Mingxiao Li, Pengcheng Chen, Yu Dong, Christopher Brinton, and Jiebo Luo. SePPO: Semi-policy preference optimization for diffusion alignment. *arXiv preprint arXiv:2410.05255*, 2024.

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

[58] 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.

[59] Mingzhe Zheng, Weijie Kong, Yue Wu, Dengyang Jiang, Yue Ma, Xuanhua He, Bin Lin, Kaixiong Gong, Zhao Zhong, Liefeng Bo, Qifeng Chen, and Harry Yang. Manifold-aware exploration for reinforcement learning in video generation, 2026. URL `https://arxiv.org/abs/2603.21872`.

[60] Zangwei Zheng, Xiangyu Peng, Tianji Yang, Chenhui Shen, Shenggui Li, Hongxin Liu, Yukun Zhou, Tianyi Li, and Yang You. Open-Sora: Democratizing efficient video production for all, March 2024. URL `https://github.com/hpcaitech/Open-Sora`.

[61] Huaisheng Zhu, Teng Xiao, and Vasant G Honavar. Dspo: Direct score preference optimization for diffusion model alignment. In *The Thirteenth International Conference on Learning Representations*, 2025.

[62] Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, Zhangwei Gao, Erfei Cui, Xuehui Wang, Yue Cao, Yangzhou Liu, Xingguang Wei, Hongjie Zhang, Haomin Wang, Weiye Xu, Hao Li, Jiahao Wang, Nianchen Deng, Songze Li, Yinan He, Tan Jiang, Jiapeng Luo, Yi Wang, Conghui He, Botian Shi, Xingcheng Zhang, Wenqi Shao, Junjun He, Yingtong Xiong, Wenwen Qu, Peng Sun, Penglong Jiao, Han Lv, Lijun Wu, Kaipeng Zhang, Huipeng Deng, Jiaye Ge, Kai Chen, Limin Wang, Min Dou, Lewei Lu, Xizhou Zhu, Tong Lu, Dahua Lin, Yu Qiao, Jifeng Dai, and Wenhai Wang. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models, 2025. URL `https://arxiv.org/abs/2504.10479`.

A Overview

This appendix provides additional training and inference details (Section B), including the data sampling procedure, training-set statistics, the decoding settings used to generate reference targets, and the training cost of all online methods; a qualitative and quantitative analysis of reward gradients (Section C); additional quantitative results, the human-evaluation protocol, and run-to-run variance (Section D); additional ablation studies covering question sets, input frames, gradient truncation, explanations, reference videos, loss normalization, and the efficient decoder (Section E); an expanded discussion of limitations (Section F) and broader impacts (Section G); further qualitative evidence, including single-prompt fine-tuning samples and gradient heatmaps (Section H); and the full prompting configurations (Section I).

B Training Details

B.1 Configuration

We provide further training and inference details in this section to support reproducibility. Overall, we follow the standard configuration of Wan2.1-1.3B-T2V [42]; Table 4 lists the VAE, text-encoder, and scheduler settings.

Table 4: Training Configure.

class	config	value
VAE	temporal_compression_ratio	4
	spatial_compression_ratio	8
Text Encoder	tokenizer	google/umt5-xxl
	text_length	512
	vocab size	256384
	dim	4096
	dim_attn	4096
	dim_ffn	10240
	num_heads	64
	num_layers	24
	num_buckets	32
	shared_pos	False
	dropout	0.0
Scheduler	num_train_timesteps	10000
	shift	5.0
	use_dynamic_shifting	false
	base_shift	0.5
	max_shift	1.15
	base_image_seq_len	256
	max_image_seq_len	4096

B.2 Data Sampling Procedure

We randomly sampled 25k video-caption pairs based on the category distribution in OpenVid-1M [29]. These pairs were then processed by our prompting pipeline to get the reference answers. To ensure the VLM could correctly understand the video, we retained only the cases where all references were 'yes'. This process resulted in our final dataset.

B.3 Training-Set Statistics and Supervision Scope

The final training set contains 5,077 caption–video pairs and 24,252 structured queries, with a TA:Phy:VQ ratio of roughly 2.5:1:1. Each OpenVid caption is used as the prompt for an on-policy sample from the generator, and its paired real video serves *only* as the reference anchor for the Phy and VQ queries during training. TA queries evaluate the generated video alone, real videos are never used as pixel- or flow-matching targets, and inference is entirely reference-free. The data requirement of Diffusion-DRF is therefore *captioned real-video pairs for reference-guided training*;

captions alone are sufficient for the TA dimension but not for Phy and VQ (see Section E.5 for the corresponding ablations). We verified that no training caption appears verbatim in the VBench-2.0 or VideoGen-Eval prompt sets. Finally, "free" in the name of the method means free of reward-model training and of preference-data collection; it does not mean free of computation (Section B.5) or of training data. The three supervision groups (TA, Phy, VQ) are operational groupings of the questions that our pipeline produces rather than a unique or complete factorization of video quality; additional groups such as long-range temporal consistency are a natural extension.

B.4 Reference-Target Generation

Reference answers and explanations are generated once, offline, with a fixed Qwen2.5-VL-7B-Instruct checkpoint [1] using `do_sample=True`, `temperature=0.1`, `top_k=1`, `top_p=0.001`, `repetition_penalty=1.05`, and `max_new_tokens=512`. Because `top_k=1` retains a single candidate per step, decoding is effectively deterministic. Explanations are constrained to fewer than 50 words by the prompt (Section I), and all targets are cached and reused across methods, ablations, and seeds, so every comparison in the paper is made against identical supervision.

B.5 Training Cost

Table 5: **Training cost of online reward-tuning methods.** All methods fine-tune Wan2.1-1.3B-T2V on the same 16×A100-80GB node; "Selected step" is the checkpoint reported in Table 1. GPU-hours are computed as GPUs × sec./step × selected step$/3600$. Flow-DPO is omitted because it trains offline on pre-collected preference pairs at a single sampled timestep and does not roll out videos during optimization.

Method	GPUs	Peak VRAM / GPU	Sec. / step	Selected step	GPU-hours	VBench-2.0
VideoAlign [24]	16×A100-80GB	72 GB	22.50	500	50.00	53.55
Flow-GRPO [23]	16×A100-80GB	68 GB	75.00	500	166.67	50.64
Diffusion-DRF (7B)	16×A100-80GB	75 GB	23.73	1500	158.17	**55.38**

Table 5 reports the GPU configuration, peak per-GPU memory, wall-clock time per optimization step, the selected checkpoint, and the resulting GPU-hours for the three online reward-tuning methods in Table 1. Per-step cost of Diffusion-DRF (23.73 s) is close to that of VideoAlign (22.50 s), since both back-propagate a differentiable reward through the last K denoising steps, and roughly three times lower than that of Flow-GRPO (75.00 s), which must roll out a group of videos per prompt. The main cost difference comes from the checkpoint selected: because Diffusion-DRF keeps improving instead of collapsing (Figure 6), its best checkpoint is at step 1500, whereas VideoAlign peaks at step 500. Consequently, the 1.83-point gain over VideoAlign in Table 1 is obtained at about 3.2× its GPU-hours, while the 4.74-point gain over Flow-GRPO is obtained at slightly lower total cost. Flow-DPO [24] trains offline on human-annotated preference pairs with a DPO-style flow-matching loss at a single sampled timestep and does not generate rollout videos during optimization, so its cost is not comparable to the online methods above and is omitted.

C Gradient Analysis

C.1 Gradient Visualization

Similar to Section 1, we further analyze the gradients induced by different reward formulations. Gradients directly determine the optimization direction. The activated regions in a gradient heatmap correspond to the parts of the video that the reward signal considers most responsible for the mismatch with the prompt. In other words, correctly localized gradient activation indicates that the reward provides semantically aligned corrective signals. We visualize and compare the gradient distributions produced by customized reward VideoAlign [24], Vanilla-DRF, and Diffusion-DRF. As illustrated in Figure 8, when fine-grained details are missing or incorrectly rendered, *only Diffusion-DRF produces spatially localized gradients that precisely highlight the regions requiring modification*. In contrast, customized reward models [24] trained on narrow preference data often produce diffuse or misplaced gradient signals, reflecting shortcut correlations or dataset bias, compared to ours that use a frozen VLM, a general large-scaled trained critic. Similarly, without multi-dimensional rich feedback,

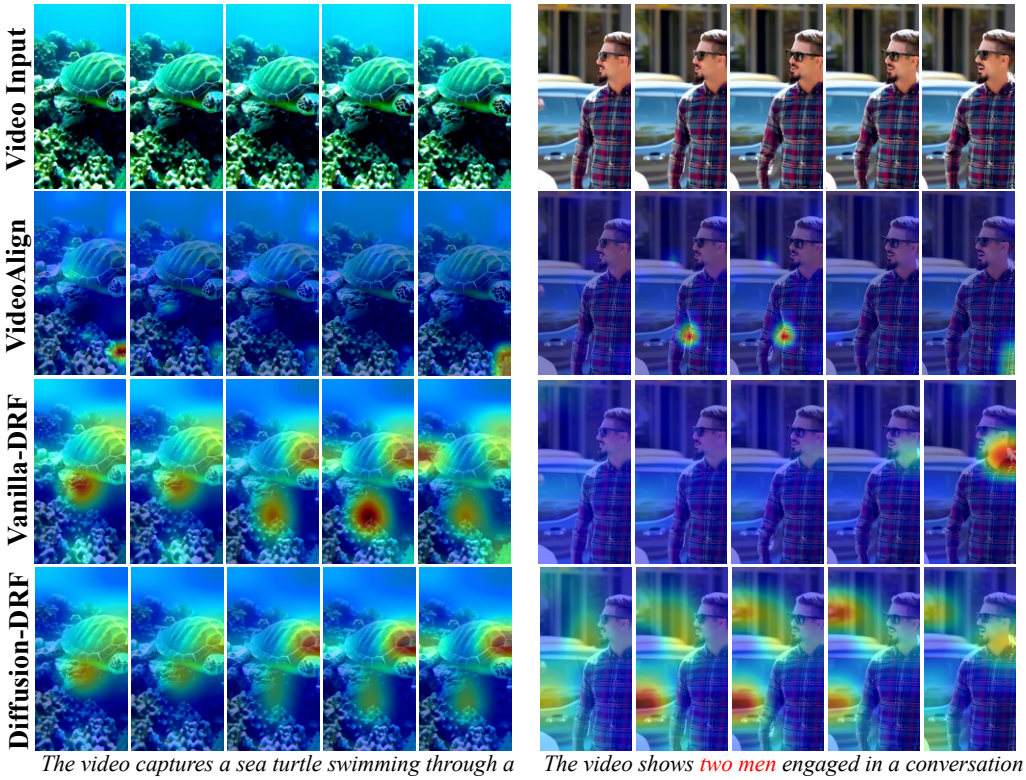


Figure 8: **Gradient heatmap visualizations for different reward formulations.** Original video failed to satisfy the color requirement of the turtle's shell (left) and two men (right).

a single global reward query in Vanilla-DRF can entangle multiple semantic factors, leading to ambiguous optimization signals. As a result, the activated regions do not consistently correspond to the true source of error in the generated video. More visualizations are shown in Figures 11 and 12.

C.2 Quantitative Gradient Localization

Table 6: **Gradient localization on synthetic defects.** Localization enrichment (higher is better) is the fraction of absolute reward-gradient mass that falls inside the defect mask divided by the mask's spatio-temporal volume fraction; $1.0\times$ corresponds to a spatially uniform gradient. Left: 200 held-out clips with localized blur, noise, or frame-freeze corruptions inserted into foreground or background regions. Right: a second set of 200 held-out clips whose captions contain no background description, split by whether the corrupted region is caption-grounded foreground or caption-unmentioned background. "Random masks" are size-, shape-, and duration-matched control masks.

Objective	All defects	Split by caption grounding	
		Unmentioned background	Caption-grounded foreground
Random masks (control)	$1.0\times$	$1.00\times$	$1.00\times$
Answer only	$1.3\times$	$1.34\times$	$1.45\times$
TA only	–	$1.05\times$	$2.34\times$
Phy only	–	$2.00\times$	$2.00\times$
VQ only	–	$2.15\times$	$2.12\times$
Full Diffusion-DRF	$\mathbf{2.5}\times$	$\mathbf{2.23}\times$	$\mathbf{2.52}\times$

The heatmaps above are illustrative, so we additionally quantify how well the reward gradient localizes defects. The reward is computed from the terminal (fully denoised) video, but a terminal objective can still produce highly non-uniform gradients over pixels and frames; the question is whether the non-uniformity points at the actual errors.

Protocol. We take 200 held-out clips and insert synthetic defects: localized blur or noise, or a short frame freeze, placed in either a foreground or a background region. Captions and cached targets are

kept fixed, and the clean clip is used as the Phy/VQ reference where applicable. We back-propagate the reward of each objective to the corrupted video and measure *localization enrichment*: the fraction of absolute gradient mass that falls inside the defect mask divided by the mask's spatio-temporal volume fraction, so that a spatially uniform gradient scores $1.0\times$. As a control we evaluate the same gradient against random masks matched in size, shape, and duration. All objectives receive identical inputs.

Results. Defects occupy about 10% of the video volume but receive 25% of the gradient mass of Diffusion-DRF, a $2.5\times$ enrichment, compared to $1.3\times$ for the answer-only objective and $1.0\times$ for random masks (Table 6, left). This demonstrates content-sensitive localization in decoded-video space; it does not measure credit assignment across denoising steps, which is studied separately through the truncation length in Section E.3.

Content not mentioned in the caption. A caption describes only part of a video, so we further ask whether the gradient can localize defects in regions that the prompt never mentions, for example a corrupted background behind "a cat is walking". We repeat the protocol on 200 held-out clips whose captions contain no background description. Caption-grounded foreground masks are produced with GroundingDINO [25] and SAM 2 [36], background masks with panoptic segmentation, both followed by manual verification. Table 6 (right) shows that the TA objective localizes almost exclusively caption-grounded content ($2.34\times$ on foreground versus $1.05\times$ on unmentioned background), whereas the VQ and Phy objectives localize visually or physically defective regions regardless of whether they are named ($2.12\times$/$2.15\times$ and $2.00\times$/$2.00\times$). Because VQ and Phy assess perceptual quality and physical plausibility rather than pixel correspondence, they can penalize a corrupted background without requiring it to match the reference background; the reference provides calibration rather than a pixel target. The full objective inherits both behaviors ($2.52\times$ foreground, $2.23\times$ background), which is consistent with the complementary roles of the three dimensions observed in Section E.1.

D Additional Results

D.1 Generalization to Another Backbone

Table 7: **Generalization to CogVideoX.** We report VBench-2.0 results of CogVideoX and of CogVideoX fine-tuned with Diffusion-DRF.

Method	*VBench-2.0*					
	Overall	Creativity	Common Sense	Controllability	Human Fidelity	Physics
CogVideoX	49.52	37.88	54.87	24.56	78.10	52.18
w/ Diffusion-DRF	51.89	40.63	57.75	27.11	81.28	52.69

We further apply Diffusion-DRF to CogVideoX [52] to evaluate its generalization beyond the main diffusion backbone. As shown in Table 7, Diffusion-DRF consistently improves the base CogVideoX model across all reported dimensions on VBench-2.0, raising the overall score from 49.52 to 51.89. Notably, the gains are not limited to a single aspect: Creativity improves from 37.88 to 40.63, Common Sense from 54.87 to 57.75, Controllability from 24.56 to 27.11, Human Fidelity from 78.10 to 81.28, and Physics from 52.18 to 52.69. These results suggest that the benefit of Diffusion-DRF is backbone-agnostic, and that the proposed differentiable reward can consistently enhance multiple aspects of video generation rather than over-optimizing a narrow subset of metrics.

D.2 Additional Pairwise Results on VideoGen-Eval

Table 2 in the main paper summarizes the pairwise comparison between Diffusion-DRF and the base model using both VideoAlign-based automatic scoring and independent human judgments. Here we provide a more detailed automatic analysis with VideoAlign. For each caption, we generate two videos with the same configuration and compare them using VideoAlign scores, where a score difference under 0.2 is counted as a tie. Figure 9 reports win / tie / loss rates for two comparisons: (i) Base vs. Diffusion-DRF and (ii) Flow-GRPO vs. Diffusion-DRF across text-video alignment, visual quality, motion quality, and overall score. Across both pairings, Diffusion-DRF attains consistently higher win rates, showing that the gains are not confined to a single metric or baseline. Because

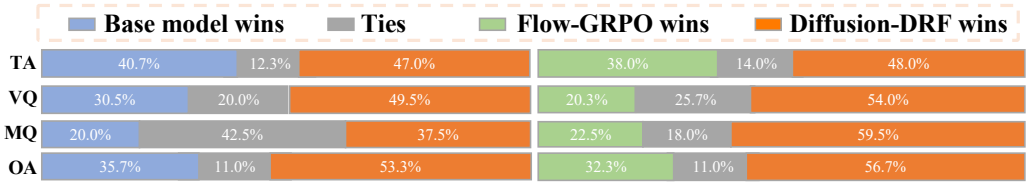


Figure 9: **Pair-wise evaluation on VideoGen-Eval.** With the same prompt and configuration, we perform pairwise comparisons of generated videos using the VideoAlign scores of text-video alignment (TA), visual quality (VQ), motion quality (MQ) and overall (OA). We compare the Diffusion-DRF with the base model (left) and the Flow-GRPO (right) respectively.

VideoAlign is itself a learned evaluator and also serves as a baseline reward, we treat this analysis as complementary per-metric evidence; the human study below is the independent validation.

D.3 Human Evaluation Protocol and Statistics

The human study in Table 2 compares Diffusion-DRF against the *base model*, the comparator for which the automatic VideoAlign preference is also reported; it therefore establishes an improvement over the base model with an evaluator that is independent of the training signal, not a ranking against the other reward-tuned baselines.

Setup. We sample 200 prompts from VideoGen-Eval [51] and generate three matched-seed pairs per prompt, giving 600 anonymized video pairs. Sixteen annotators each evaluated all 600 pairs, yielding 9,600 judgments. Model identities were hidden and the left/right order was randomized per pair. Annotators were instructed to judge prompt fidelity, namely whether the entities, attributes, counts, actions, interactions, and spatial-temporal relations described in the prompt appear in the video, and to select a tie when neither video was clearly superior. The win / tie / loss rates in Table 2 aggregate these judgments.

Statistics. Humans prefer Diffusion-DRF in 29.0% of comparisons and the base model in 17.0%, a net preference of $+12.0$ points with a 95% confidence interval of $[+5.0, +19.0]$ obtained by clustering on both prompts and annotators. The large tie rate (54.0%) reflects that many prompts are already handled well by the base model, in which case Diffusion-DRF preserves the generation (Section H). Restricted to decisive comparisons, Diffusion-DRF is preferred in 63.0% of cases (95% CI $[56.0\%, 70.0\%]$), close to the 59.9% obtained under VideoAlign ($53.3/(53.3 + 35.7)$), which argues against the automatic preference being an artifact of evaluator–critic correlation. Krippendorff's α across annotators is 0.72, indicating substantial inter-annotator agreement.

On evaluator independence. Our optimization signal comes from Qwen2.5-VL, and both evaluation suites include learned components: VBench-2.0 combines generalist VLM evaluators with specialist metrics calibrated on human annotations, and VideoGen-Eval uses VideoAlign. Neither shares our critic, but both are learned, so we regard the human study as the primary independent evidence and the automatic metrics as supporting evidence.

D.4 Run-to-Run Variance

Figure 6 reports complete training trajectories with the same best-checkpoint protocol applied to every method, and Diffusion-DRF remains above all competing methods over a broad interval rather than at a single favorable checkpoint. To quantify seed variation, we repeat the default Diffusion-DRF (7B) run with three independent seeds and obtain VBench-2.0 Overall scores of 55.38, 55.72, and 55.22, i.e. 55.44 ± 0.26 (mean $\pm$ standard deviation). Even the lowest run exceeds VideoAlign (53.55) by 1.67 points and the base model (52.99) by 2.23 points. We present this as evidence of low run-to-run variation of our method; it is not a significance test against baselines for which only a single run is available, and throughout the appendix we treat differences within about one standard deviation (± 0.26) as not measurable.

Table 8: **Detailed results of ablation studies on question sets and styles.**

Method	*VBench-2.0*					
	Overall	Creativity	Common Sense	Controllability	Human Fidelity	Physics
Baseline	52.99	53.79	55.52	26.59	**80.65**	48.40
TA Only	52.15	49.41	53.80	27.03	74.88	55.64
VQ Only	52.45	50.53	53.22	27.14	80.22	51.28
Phy Only	51.81	37.04	57.82	27.14	79.46	<u>56.28</u>
TA + Phy	53.55	50.56	**60.12**	25.72	75.57	55.78
TA + VQ	53.92	<u>52.34</u>	55.81	<u>27.80</u>	79.02	54.65
VQ + Phy	53.20	44.50	55.81	27.10	79.02	55.64
TA + VQ + Phy	**55.38**	**54.58**	<u>56.96</u>	**27.98**	<u>80.51</u>	**56.85**
Binary question	53.30	50.56	56.96	25.72	75.57	55.78
Explanation w/ CoT	54.85	53.75	59.39	27.52	79.53	56.72

E Additional Ablation Studies

E.1 Question Sets

We provide the detailed results of the ablation studies of the question sets in Table 8. Consistent with the summary in the main paper, all single-dimension settings underperform the baseline, all two-dimension combinations surpass the baseline, TA+VQ is the strongest two-way combination, and TA+VQ+Phy gives the best overall result. The per-dimension scores further clarify how each feedback type shapes model behavior. TA focuses on explicit semantic entities, attributes, and actions, so it improves *Controllability* from 26.59 to 27.03 and substantially raises *Physics* from 48.40 to 55.64, but still hurts *Creativity* (49.41), *Common Sense* (53.80), and *Human Fidelity* (74.88), lowering *Overall* from 52.99 to 52.15: TA rewards the requested entities, attributes, and actions but does not explicitly penalize perceptual artifacts or distorted human appearance, so alignment-only supervision is still easy to exploit. VQ emphasizes perceptual quality, artifacts, and appearance consistency, and therefore best preserves *Human Fidelity* among the single-dimension settings at 80.22 while modestly improving *Controllability* to 27.14; however, its gains on *Creativity* (50.53), *Common Sense* (53.22), and *Physics* (51.28) remain limited. Phy targets motion realism and physical plausibility, which explains why it gives the strongest single-dimension *Physics* score of 56.28 and also improves *Common Sense* to 57.82, but it severely hurts *Creativity* to 37.04, showing that physical supervision alone can over-regularize the generator.

Combining dimensions leads to much more balanced behavior. TA+Phy reaches the highest *Common Sense* score in the whole table at 60.12 and maintains strong *Physics* at 55.78, but its *Controllability* (25.72) and *Human Fidelity* (75.57) remain weak. TA+VQ is the strongest two-way combination in terms of *Overall* score at 53.92, and its gains come from a better balance across *Creativity* (52.34), *Controllability* (27.80), and *Human Fidelity* (79.02), confirming the main-paper observation that visual-quality supervision is particularly important for stabilizing alignment-oriented optimization. VQ+Phy is also better than the baseline, with an *Overall* score of 53.20, but it remains below TA+VQ, suggesting that physical and perceptual feedback alone are still insufficient without explicit semantic alignment constraints. Finally, TA+VQ+Phy gives the best *Overall* score of 55.38 and the strongest or near-strongest values on all major axes, including *Creativity* (54.58), *Controllability* (27.98), *Human Fidelity* (80.51), and *Physics* (56.85). This result supports our central claim that the three dimensions provide complementary constraints: TA anchors prompt semantics, VQ protects perceptual quality, and Phy improves motion and physical plausibility.

E.2 Number of Input Frames

We study how the number of video frames fed into the VLM affects training. Generated videos contain 49 frames; N_f denotes the number of uniformly subsampled frames given to the critic (and, for Phy/VQ, to the reference), not the video length. As shown in Table 9, increasing N_f leads to a clear and consistent improvement in overall performance, with the score rising from 50.83 at $N_f = 2$ to 55.38 at $N_f = 10$; the last step from $N_f = 8$ to $N_f = 10$ still adds 0.58 points (54.80 to 55.38) at the cost of 4 GB of memory (71 GB to 75 GB). This trend indicates that denser temporal observations help the VLM better understand the video content and provide more informative reward signals. In particular, larger N_f also improves several key dimensions, including Common Sense and Physics,

Table 9: **Additional ablation studies on input frames.** We report the results of the ablations on changing the number of input frames for the VLM on VBench-2.0. All models are trained with the same configuration as Diffusion-DRF.

Method	*VBench-2.0*						Training VRAM Usage (Gb)
	Overall	Creativity	Common Sense	Controllability	Human Fidelity	Physics	
$N_f = 2$	50.83	48.43	48.10	27.50	80.96	54.14	62
$N_f = 4$	51.45	43.22	53.51	27.64	78.4	54.45	65
$N_f = 6$	52.19	43.85	55.23	29.25	78.88	53.75	68
$N_f = 8$	54.80	54.50	55.81	28.10	79.61	56.03	71
$N_f = 10$	55.38	54.58	56.96	27.98	80.51	56.85	75
$K = 4$	55.38	54.58	56.96	27.98	80.51	56.85	75
$K = 3$	55.05	52.75	58.11	27.52	80.80	56.09	69
$K = 2$	54.36	52.45	57.24	27.45	78.50	56.18	63
$K = 1$	53.16	52.95	52.93	25.45	79.49	54.99	56
CoT (1K-steps)	54.25	52.75	58.11	26.52	79.80	56.09	75
CoT (2K-steps)	54.85	53.75	59.39	27.52	79.53	56.72	75
CoT (3K-steps)	51.89	43.75	55.81	24.83	79.13	53.92	75
w/o ref frames	53.15	50.41	54.50	27.80	78.88	56.34	72

suggesting that richer temporal coverage is especially helpful for reasoning about object dynamics and physical plausibility. For these 49-frame clips we observe that the critic's responses remain largely unchanged when N_f exceeds 10, suggesting that the feedback saturates at the current duration, so we use $N_f = 10$ as the coverage–memory trade-off. This saturation should not be extrapolated to longer videos, where sparse sampling may miss multi-stage actions or delayed physical effects (Section F).

E.3 Gradient Truncation Length

We set the number of back-propagation steps K to 4 in the main results and ablate its effect in Table 9. Back-propagating only through the final K denoising steps yields a biased pathwise gradient estimator of the full-chain gradient; we make no claim of unbiasedness or of equivalence to full-chain back-propagation, and evaluate the truncation empirically. Using more back-propagation steps consistently improves performance: the overall score increases from 53.16 at $K = 1$ to 54.36, 55.05, and 55.38 at $K = 2$, 3, and 4, while peak memory grows from 56 GB to 75 GB. The marginal gains diminish, from $+1.20$ for $K{=}1 \rightarrow 2$ to $+0.69$ for $K{=}2 \rightarrow 3$ and $+0.33$ for $K{=}3 \rightarrow 4$, the last of which is close to the run-to-run standard deviation (Section D.4). This suggests that propagating reward gradients through more denoising steps provides stronger corrective supervision, with most of the benefit obtained within the last few steps; larger K also incurs higher training cost and memory overhead, which limits further scaling in practice.

E.4 Explanations: Length, Ordering, and Content

Table 10: **Explanation ordering and content controls (VBench-2.0 Overall).** All variants use the same queries, data, and optimization settings and differ only in the supervised target sequence. The shuffled control keeps the answer tokens and the explanation length but pairs each query with an explanation drawn from another sample in the same supervision group with the same answer label. Three-seed statistics are reported where available (Section D.4).

Target format	Single run	Three seeds (mean $\pm$ std)
Answer only	53.30	–
Shuffled explanation (length-matched)	52.47	–
Explanation $\rightarrow$ Answer	55.24	55.31 ± 0.16
Answer $\rightarrow$ Explanation (default)	**55.38**	$\mathbf{55.44 \pm 0.26}$

Explanation length. By default, we use reference explanations with an average length of 50 tokens. We also adopt Chain-of-Thought (CoT) prompting to generate longer reference explanations ("Explanation w/ CoT" in Table 8), which typically expands the explanation to over 100 words and introduces much more redundant content. As shown in Table 9, CoT supervision can improve performance at the early stage of training, achieving 54.85 at 2K steps, which suggests that longer

explanations initially provide richer diagnostic cues. However, this benefit is short-lived: performance drops to 51.89 at 3K steps, indicating substantially earlier collapse than the non-CoT setting. We attribute this to the excessive verbosity of CoT outputs, where many irrelevant tokens dilute useful gradients and amplify optimization noise during training.

Explanation ordering. In our default target format the Yes/No token precedes the explanation, so the explanation is generated conditioned on an already fixed decision and cannot causally inform it. We therefore describe the explanation as *rationale supervision* rather than chain-of-thought reasoning: its role is to provide additional target tokens whose likelihoods, through the frozen VLM, are differentiated with respect to the generated video. To test whether the order matters, we train a variant whose targets place the explanation before the answer, holding data, targets, and optimization fixed (Table 10). Explanation-first obtains 55.24 versus 55.38 for answer-first in a single run, and 55.31 ± 0.16 versus 55.44 ± 0.26 over three matched seeds; the paired mean difference of 0.13 points (95% CI $[-0.12, 0.38]$) provides no evidence of an ordering effect. This is expected, since both formats contain the same target tokens and ordering only changes the autoregressive conditioning, not the supervision content. The meaningful contrast is between either explanation-containing format (55.31/55.44) and the answer-only objective (53.30).

Explanation content. Longer targets alone could conceivably help simply by providing more tokens. We rule this out with a length-matched control that shuffles explanations across samples within the same supervision group and answer label, so that each query is supervised with a fluent but unrelated explanation. This control obtains 52.47, *below* the answer-only objective (53.30) and below the base model, showing that the gain from explanations comes from their semantics rather than from the number of target tokens, and that mismatched rationales actively harm the reward signal.

E.5 Role of the Reference Video

Table 11: **Role of the caption-matched reference video (VBench-2.0).** "Shuffled reference" pairs each caption with a real video from a different caption; "SFT" directly fine-tunes the generator on the same 5K real videos with a matched update budget. Diversity is the VBench-2.0 diversity score (higher means more cross-sample variation).

Setting	Overall	Diversity
Base model	52.99	58.6
PickScore [19]	49.62	29.5
VideoAlign [24]	53.55	52.4
SFT on reference videos (matched budget)	53.42	–
Diffusion-DRF, no reference	53.15	–
Diffusion-DRF, shuffled reference	53.06	–
Diffusion-DRF, matched reference	**55.38**	**64.4**

Table 12: **Reference conditioning versus explanation supervision (VBench-2.0 Overall).** A 2×2 factorial over (i) whether Phy/VQ queries are conditioned on the caption-matched reference video and (ii) whether free-form explanations are supervised in addition to the Yes/No answers, on two training corpora. "Answer only" and "Full" use the same TA/VQ/Phy query set. The VideoUFO column uses 5,000 caption–video pairs from VideoUFO [43] with the same filtering, query pipeline, TA:Phy:VQ ratio, initialization, and optimization settings. Rows within ±0.26 of the base model (one run-to-run standard deviation, Section D.4) are not distinguishable from it.

Training setting	Reference	Explanation	OpenVid-1M	VideoUFO
Base model (Wan2.1-1.3B-T2V)			52.99	52.99
Answer only, no reference			53.02	53.10
Full feedback, no reference		✓	53.15	53.22
Answer only, with reference	✓		53.30	53.54
Full Diffusion-DRF	✓	✓	**55.38**	**55.15**
Interaction (difference-in-differences)			+1.95	+1.49

Phy and VQ queries condition the frozen critic on a caption-matched real video (Section 3.2). Since the generator is trained on on-policy samples and the reference receives no gradient, Diffusion-DRF is not a form of supervised fine-tuning on real videos, but the reference could still bias the generator toward the reference distribution. We examine both questions.

Matched references are required. Table 11 compares the default model with a no-reference variant (Phy/VQ queries evaluate the generated video alone) and a shuffled-reference variant (each caption is paired with a real video from a different caption). Removing references reduces VBench-2.0 Overall from 55.38 to 53.15 and shuffling them gives 53.06; both are within one run-to-run standard deviation of the base model (52.99, Section D.4). We therefore do not claim a measurable improvement without matched references: the gain of Diffusion-DRF is contingent on reference-calibrated Phy/VQ feedback, and mere exposure to real videos is not sufficient.

Diffusion-DRF is not reference imitation. Directly fine-tuning the generator on the same 5K real videos with a matched update budget (SFT) obtains 53.42, well below Diffusion-DRF (55.38), and Diffusion-DRF *increases* VBench-2.0 Diversity from 58.6 for the base model to 64.4, whereas the learned scalar rewards reduce it (52.4 for VideoAlign and 29.5 for PickScore, consistent with the prompt-unrequested stylization visible in Figures 5, 7 and 10). Diversity measures cross-sample variation rather than style bias directly, so we use it only as supporting evidence.

Reference conditioning and explanation supervision interact. To separate the contribution of references from that of the rich feedback, Table 12 reports a complete 2×2 factorial over reference conditioning and explanation supervision, on OpenVid-1M and on an independently collected corpus of 5,000 VideoUFO [43] caption–video pairs processed with the same filtering pipeline (OpenVid source overlaps and caption/visual near-duplicates removed) and identical query pipeline, TA:Phy:VQ ratio, initialization, and optimization. Under answer-only supervision, references add only $+0.28$/$+0.44$ points (OpenVid/VideoUFO); without references, explanations add only $+0.13$/$+0.12$. With full feedback, references add $+2.23$/$+1.93$, and with references fixed, explanations add $+2.08$/$+1.61$. The difference-in-differences interaction is $+1.95$/$+1.49$, substantially larger than either isolated effect, on both corpora. Neither component alone reproduces the full gain: references calibrate the Phy/VQ judgments, and the rich token-level feedback converts the resulting multi-dimensional judgments into gradients on on-policy videos. Because the two corpora have different native captions we do not interpret their absolute-score difference as a controlled corpus effect; the matched within-corpus comparisons show that the interaction transfers to a second corpus.

E.6 Loss Normalization

For brevity the objective in Section 3.2 is written as a sum over queries and tokens. The implementation normalizes it as a mean over valid target tokens,

$$\mathcal{L}_{\text{VQA}} = -\frac{\sum_{i=1}^{N}\sum_{t=1}^{L_i} m_{i,t}\,\log p_\phi\Big(y_i^{(t)} \mid \mathbf{c}_i, y_i^{(<t)}\Big)}{\sum_{i=1}^{N}\sum_{t=1}^{L_i} m_{i,t}}, \tag{4}$$

where $m_{i,t} \in \{0,1\}$ masks padding tokens, so that the effective learning rate does not scale with the number or length of the queries of a prompt. Because prompts of different complexity yield different numbers of TA queries, we also evaluate a per-query mean (each query contributes equally) and a per-dimension-balanced mean (TA, Phy, and VQ contribute equally): they obtain 55.31 and 55.45 versus 55.38 for the valid-token mean. The three schemes are within run-to-run variance of each other, showing that the gain is not driven by response-length or dimension-frequency weighting; we keep the valid-token mean in all reported results.

E.7 Efficient Video Decoder

The H3AE decoder [46] used in the reward path (Section 4.1) sits inside the differentiable chain, so we verify that it does not change the reward signal. It is a lighter decoder trained to reproduce the reconstructions of the original Wan2.1 VAE decoder and is used solely to reduce memory; it is not a source of quality improvement. At $K = 2$, training with the original decoder obtains 54.7 VBench-2.0 Overall versus 54.4 with H3AE, a difference within one run-to-run standard deviation (Section D.4), while peak memory increases from 63 GB to 81 GB. H3AE reduces the memory of the decoding step by 24.3%, which is what makes $K = 4$ and $N_f = 10$ fit on 80 GB GPUs.

F Limitations and Future Work

Paired training data. Although Diffusion-DRF needs neither a reward model nor preference annotations, its Phy and VQ dimensions require a caption-matched real video for every training

prompt (Section B.3), and the gain is contingent on matched references (Section E.5). Domains without such a paired corpus would need one to be collected; extending the reference-free variant, for example with retrieved rather than paired references, is an open direction.

Training–evaluation coverage. The model is tuned on 5K OpenVid-1M videos and evaluated on VBench-2.0 and VideoGen-Eval prompts. We verified that no training caption appears in the evaluation sets and that the reference/explanation interaction transfers to a second corpus (VideoUFO), but we have not characterized how far the evaluation prompts lie from the training distribution, and coverage of the training captions, rather than their number, likely bounds the improvement.

Evaluator correlation. The critic is a VLM and both benchmarks contain learned evaluators. The human study (Section D.3) provides independent evidence against the base model only; human comparisons against the strongest reward-tuned baselines remain to be done.

Sensitivity to the wording of cached explanations. The objective maximizes the likelihood of a single cached reference explanation. Natural language admits many equivalent phrasings, so a correct video whose content would elicit a differently worded explanation is penalized; caching makes comparisons reproducible but does not provide semantic invariance. Multi-paraphrase targets or semantic-matching objectives are natural remedies.

Approximations in the differentiable path. Truncated back-propagation is a biased estimator whose bias we only characterize empirically (Section E.3), the H3AE decoder approximates the original VAE decoder (Section E.7), and the critic sees $N_f = 10$ uniformly sampled frames. The last choice is adequate for 49-frame clips but may miss multi-stage actions or delayed physical effects in longer videos, so long-horizon scalability is untested.

Critic scale and run-to-run variance. Our critics are capped at 7B parameters; the monotonic trend from 1B to 3B to 7B (53.10, 53.72, 55.38) is established only within this range. Three-seed variance is reported for our method (Section D.4) but the baselines are single runs, so small margins between baselines should not be over-interpreted.

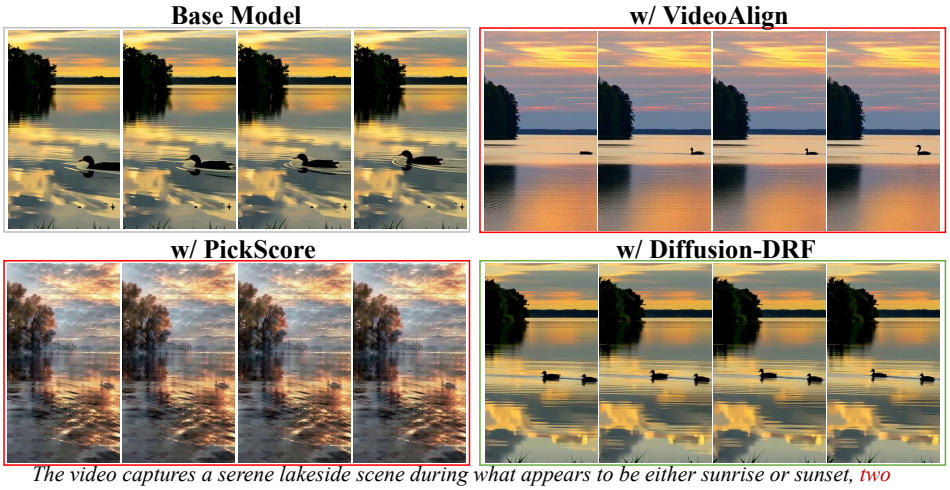
Training stability and cost. Video-generation alignment tuning remains challenging. Prior reward-tuning approaches [59, 49, 24] can empirically be trained for fewer than 200 steps before collapsing, while Diffusion-DRF, to the best of our knowledge, is the first to remain stable up to 3K steps, although it still collapses afterward. A promising direction is to incorporate stabilization strategies from GRPO-style optimization, such as reward clipping, to further improve long-horizon training stability. Alignment tuning also requires a full-trajectory rollout of 25 denoising steps per training sample, and the longer schedule that our stability permits means that the gain over VideoAlign comes at roughly 3.2$\times$ its GPU-hours (Section B.5). Future work may explore step-distilled generators and quantized rollout (INT4/INT8) [53] to further accelerate reward tuning for video generation.

G Broader Impacts

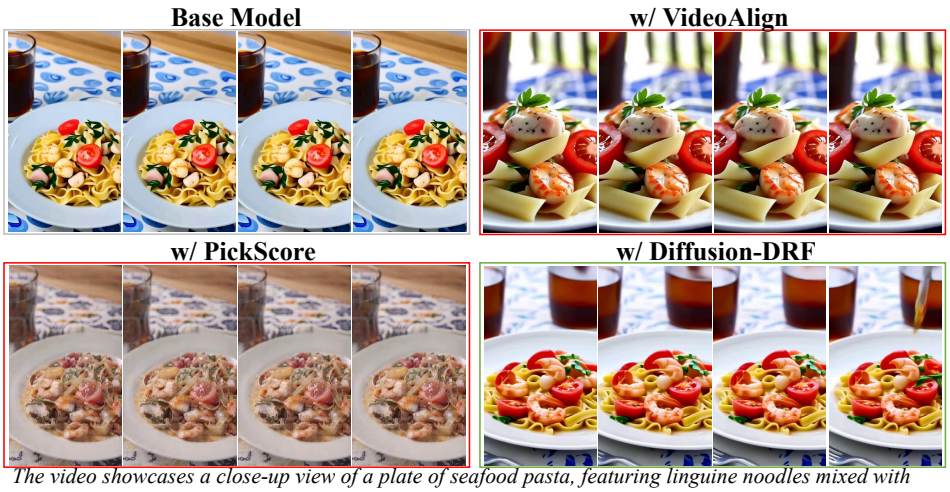
Diffusion-DRF improves the prompt adherence, physical plausibility, and visual quality of text-to-video generators without collecting human preference data, which lowers the cost of aligning generators to new requirements. The same property can lower the barrier to producing realistic but misleading video content. Our method does not add capabilities for identity synthesis or for removing provenance signals, but it also does not add safeguards of its own; fine-tuned checkpoints should be released under the base model's license and usage policy and, where possible, accompanied by safeguards such as content filters and watermarking. Because the critic is a frozen VLM, its biases (for example, toward particular visual styles or demographics) can be transferred to the generator; the reference-guided Phy/VQ queries and the human study in Section D.3 only partially control for this, and auditing the critic's judgments for such biases is an important direction. The human study collected anonymized pairwise preferences over generated videos and no personal data about the annotators.

H Additional Visual Results

Figure 10 shows more examples of single prompt fine-tuning. Figures 11 and 12 show further visualization of the gradient heatmaps. We also provide visual results in video format on the project page. On the project page, we first present qualitative comparisons among the base model, the



The video captures a serene lakeside scene during what appears to be either sunrise or sunset, two ducks are seen across the frame..



The video showcases a close-up view of a plate of seafood pasta, featuring linguine noodles mixed with shrimp, squid, and chunks of chicken. A fork is also visible on the left side of the frame.

Figure 10: **Additional Samples of Single Prompt Finetuning.**

model fine-tuned with VideoAlign, and our Diffusion-DRF model. These visual results show that our method improves both text-video alignment and physical fidelity. We also provide examples demonstrating that when the base model already produces high-quality videos, our method preserves these strengths. In contrast, VideoAlign may lead to over-optimization and deteriorate originally high-quality generations. We additionally present single prompt fine-tuning videos and several cases in which our method performs particularly well.

I Detailed Prompts

In this section, we present the prompts used in our prompting pipeline (Figures 13 and 14) as well as the prompts used during training (Figures 15 to 17), which cover the various facets used for analyzing the generated videos.

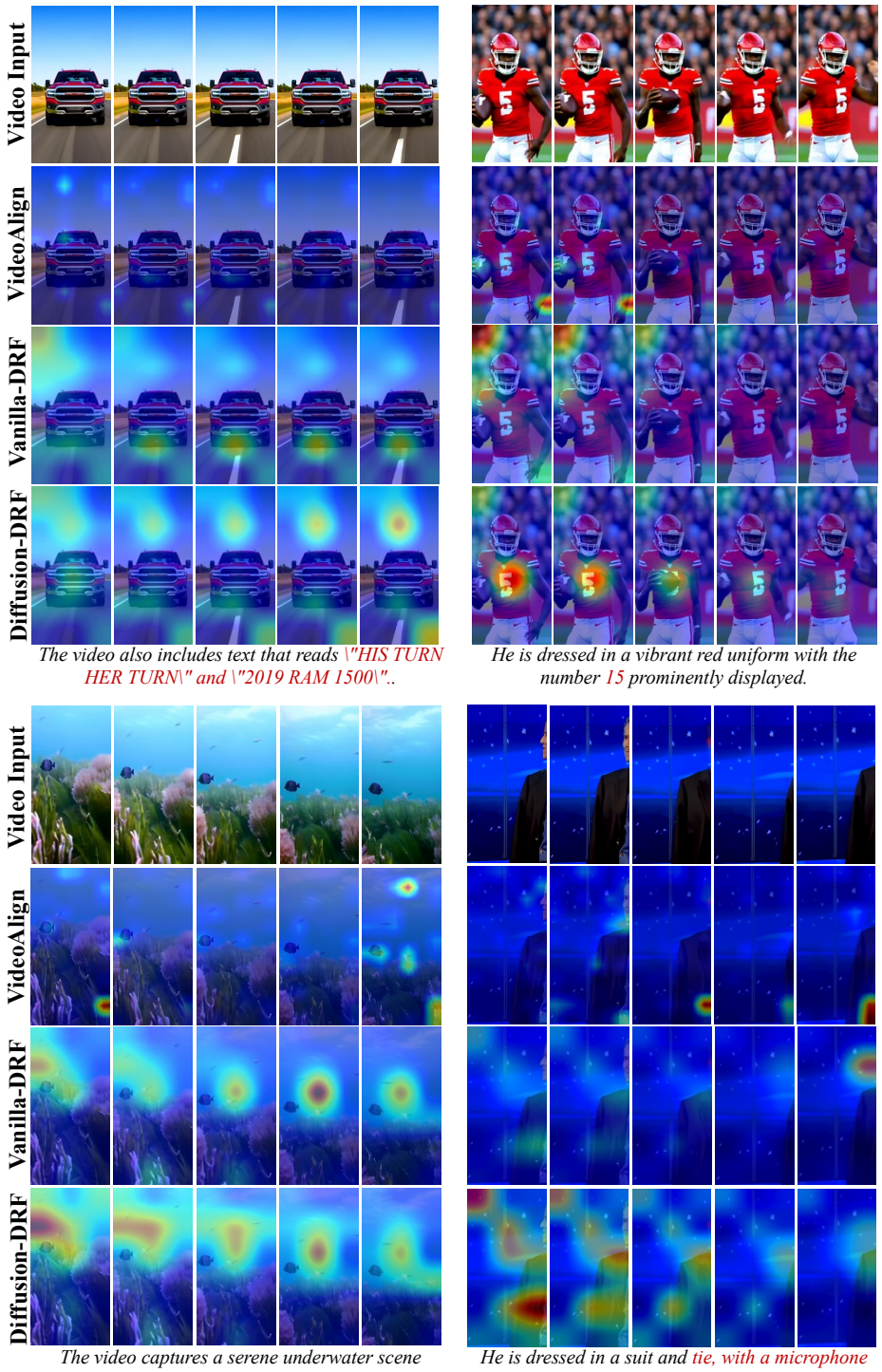


Figure 11: **Additional Gradient Heatmaps - Part1.**

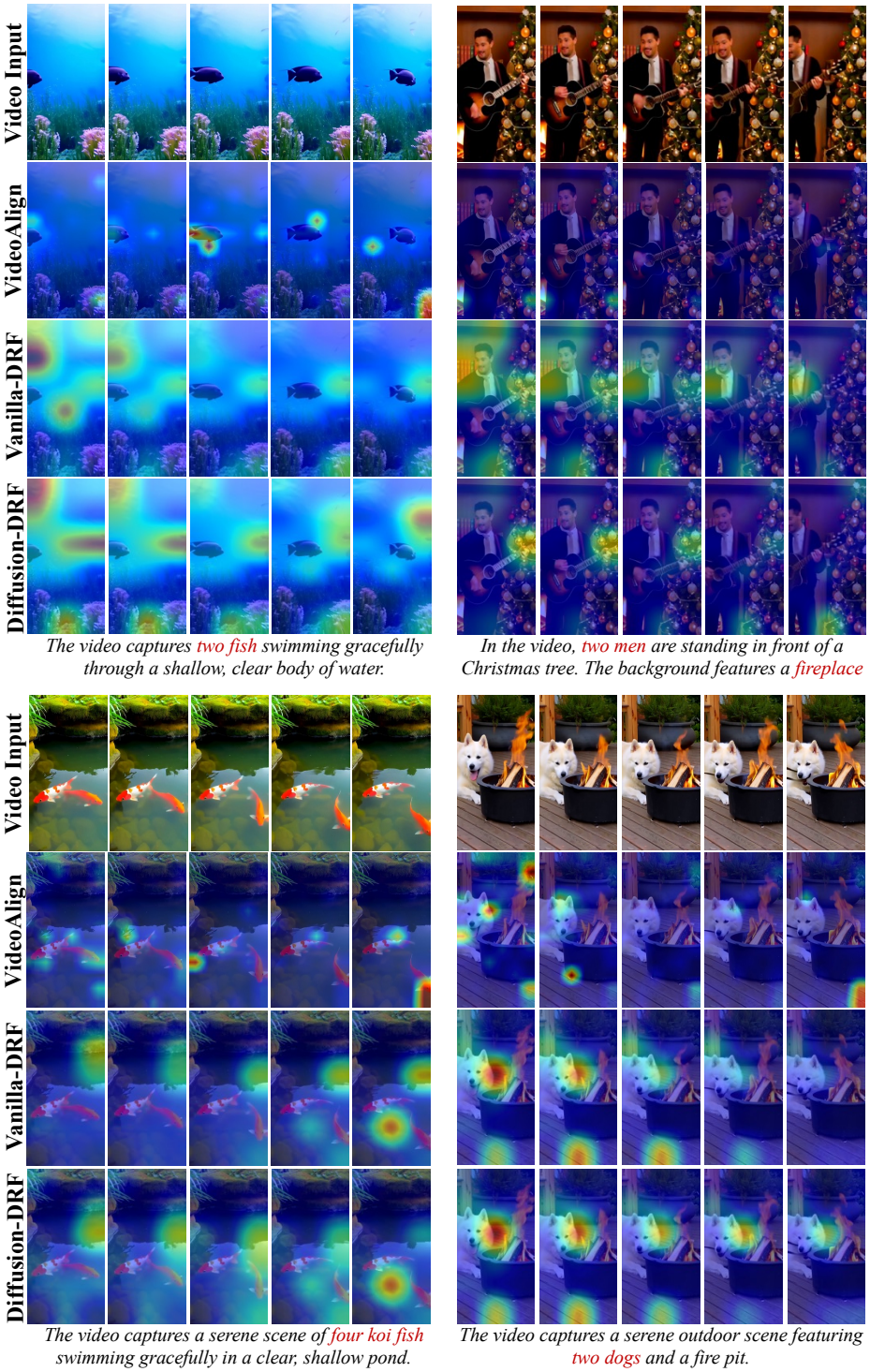


Figure 12: **Additional Gradient Heatmaps - Part2.**

Prompts of caption decomposition - Part1:
Please analyze the following video generation prompt by breaking it down based on the following key components:
1. Environment: Describe the overall setting and static elements of the environment.
2. Object(s): List the main objects/entities in the scene.
3. Objects' Motion: Describe how each object moves or interacts within the environment. Include direction, speed, and behavior.
4. Object Location / Spatial Distribution: Describe the spatial layout of objects in the scene. Where are they located (e.g., left/right/center/far/near)? Is the composition symmetrical or unbalanced?
5. Color Requirement: Describe any mentioned or implied colors for objects or environment. Mention the dominant tone or palette.
6. Lighting: Describe the light source(s), brightness, shadows, and general mood (e.g., backlit, dim, dramatic, diffuse sunlight).
7. Letter/Text Presence: Indicate whether there are any textual elements (e.g., signs, billboards), whether text is legible, and how it integrates into the scene.
8. Camera Motion: Describe how the camera moves or stays still (e.g., tracking, panning, zooming, handheld, fixed). Also indicate how this affects perception of the scene. Return the result in a structured bullet-point format, your response of each elements should be as simplified as possible.

Important: Only extract the the element is merely described atmospherically or implicitly observed — only count it if there is a clear instruction or strong implication to generate it.
Here is the prompt to analyze:
{video caption}
The answer format should be in dict format:
"Environment": content, "Object(s)": content, "Object Location/Spatial Distribution": content, "Objects' Motion": content, "Color Requirement": content, "Camera Motion": content
Here is a decomposed example for you:
The video generation prompt is
"a moment on a rainy day in a city. The street, slick with rain, reflects the surrounding buildings and trees, creating a mirror-like surface. Two motorbikes, one blue and the other white, are making their way down this wet road. They are moving towards the camera, their tires kicking up droplets of water. On the left side of the street, several tents and umbrellas have been set up, providing shelter from the rain. These structures add a splash of color to the otherwise gray scene. On the right side of the street, a red and yellow sign stands out, although the text on it is not visible. The sky overhead is a blanket of gray, heavy with rain clouds. Despite the inclement weather, there's a certain tranquility to the scene. It's as if time has slowed down, allowing one to fully take in the details of this rainy day in the city."

Figure 13: Prompts used for the caption decomposition. Part 1

Prompts of caption decomposition - Part2:
Your answer:
{
'Environment': '1). A city street during a rainy day. 2). The road is slick with rain, creating a mirror-like reflection of surrounding buildings and trees. 3). The sky is gray and heavy with rain clouds.'

'Object(s)': '1). Two motorbikes. 2). Several tents and umbrellas. 3). A red and yellow sign (text not visible).'
'Objects' Motion': '1). The two motorbikes are moving toward the camera. 2). Their tires are kicking up water droplets from the wet road. 3). Other objects (tents, umbrellas, sign) are static.'
'Object Location / Spatial Distribution': '1).The motorbikes are on the street. 2). Tents and umbrellas are placed on the left side of the street. 3). The sign is on the right side.'
'Color Requirement': '1). The overall palette is gray, reflecting the rainy weather. 2). The motorbikes are blue and white. 3). The sign is red and yellow. 4). The tents and umbrellas are colorful.'
'Lighting:': 'Not explicitly stated.'
'Letter/Text Presence': 'Not explicitly stated.'
'Camera Motion': 'Not explicitly stated.'
}

Figure 14: Prompts used for the caption decomposition - Part2.

Prompts of TA:
Given this AI-generated video, does it successfully fulfill the {key} condition: {description}? Respond with 'Yes' or 'No', Answer 'Yes' if the video largely matches the description. Answer 'No' if the video clearly contradicts the description. The presence of additional elements in the video is acceptable as long as they do not conflict with the core description. Please provide a brief explanation for your answer.
Provide your analysis and explanation in JSON with keys: answer (e.g., Yes or No), explanation.

Figure 15: Prompts used in the training for text-video alignment (TA).

Prompts of Phy:
You are a careful video forensics assistant. You are given two videos: a test video (the first 10 frames) which is ai-generated and a real video (the last 10 frames). The test video is generated by the caption of the real video. The caption is
{video prompt}
Your task is evaluating whether the test video shows physics-related defects. You should compared both videos and use the provided caption as high-level intent. Focus on physical plausibility, not style or aesthetics.
You need to analyze it in these aspects:
- Liquid flow irregularity – e.g., non-inertial or discontinuous flow, volume popping, gravity-inconsistent motion, impossible splashes.
- Abnormal object deformation – e.g., rigid objects bending/stretching without cause, topology changes (parts merging/splitting).
- Abnormal texture/material change – e.g., surface turns matte→glossy without cause, texture swimming/flicker detached from geometry.
- Abnormal motion – e.g., inertia/acceleration violations, teleporting, time reversals, jitter not explained by camera motion.
- Unnatural interpenetration – e.g., objects passing through each other or ground, missing collisions/contacts.
If the prompt doesn't have related physical aspects, you should answer 'No'.
Output strictly as JSON with this schema (no extra text):
{
"liquid flow irregularity": "Yes or No",
"abnormal deformation": "Yes or No",
"abnormal texture change": "Yes or No",
"abnormal motion": "Yes or No",
"unnatural interpenetration": "Yes or No",
}

Figure 16: Prompts used in the training for physical fidelity (Phy).

Prompts of VQ:
Compare the test video (the first 10 frames) to the reference frame (the last 10 frames) and decide if the video shows obvious visual-quality (VQ) defects relative to the reference.
Consider only: blur (defocus/motion), compression artifacts (blocking/ringing/mosquito), noise/grain, banding, flicker, rolling-shutter, aliasing/moire, over-smoothing.
Return ONLY this JSON (no timestamps):
{
"has obvious defect": Yes / No,
"dominant issue": "none / defocus blur / motion blur / blocking / ringing / mosquito noise / grain noise / banding / flicker / rolling shutter / aliasing / moire / over or smoothing",
"evidence": ["short visual cues vs reference, e.g., softer edges than reference', block edges visible around text'"],
}
Rules:
- Compare against the reference frame's look (sharpness, texture, edges, tones).
- Be conservative; if unsure, choose false and note 'uncertain' in evidence."'

Figure 17: Prompts used in the training for visual quality (VQ).

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: The abstract and introduction state the method, the absolute VBench-2.0 scores against the base model and the strongest baseline, and the scope of the evaluation; the main assumptions, notably the need for caption-matched reference videos, are stated in Section 3.3 and the Limitations paragraph of Section 5.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: The conclusion contains a Limitations paragraph (Section 5), and Section F discusses each limitation in detail.

   Guidelines:

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

3. **Theory assumptions and proofs**

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

Answer: [N/A]

Justification: We don't have any theoretical results.

Guidelines:

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

4. **Experimental result reproducibility**

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

Answer: [Yes]

Justification: The paper provides the model setup, training details, evaluation benchmarks, ablations, and supplementary implementation details needed to reproduce the reported results.

Guidelines:

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

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

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

   Answer: [Yes]

   Justification: We will release the code.

   Guidelines:

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

6. **Experimental setting/details**

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

   Answer: [Yes]

   Justification: See Section 4.1 and Section B, which cover the configuration, data sampling, training-set statistics, and reference-target decoding settings.

   Guidelines:

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

7. **Experiment statistical significance**

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

   Answer: [Yes]

   Justification: We report the mean and standard deviation of three independent runs of our method (Section D.4), a prompt- and annotator-clustered 95% confidence interval and inter-annotator agreement for the human study (Section D.3), and a paired confidence interval for the explanation-ordering comparison (Section E.4). Baselines are single runs, which we state explicitly.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

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

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: Section 4.1 gives the hardware, and Section B.5 reports peak memory, time per step, selected step, and GPU-hours for our method and the online baselines.

   Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: We reviewed the NeurIPS Code of Ethics and, to the best of our knowledge, the research conforms to it.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [Yes]

    Justification: Section G discusses the positive impact of cheaper alignment, the risk of misleading video content, and the transfer of critic biases to the generator.

Guidelines:

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

11. **Safeguards**

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

    Answer: [No]

    Justification: Section G discusses misuse risks and recommends release safeguards, but the paper does not describe a specific safeguard mechanism of its own.

    Guidelines:

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

12. **Licenses for existing assets**

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

    Answer: [Yes]

    Justification: The paper uses existing pretrained models and publicly available datasets, and these assets are cited in the manuscript.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.

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

13. **New assets**

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

    Answer: [N/A]

    Justification: The paper does not release new assets.

    Guidelines:

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

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

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

    Answer: [No]

    Justification: Section D.3 describes the task, the instructions given to annotators, the number of annotators and judgments, and the blinding and randomization; compensation details are not reported.

    Guidelines:

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

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

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

    Answer: [No]

    Justification: The paper includes a human evaluation, but the current submission does not describe participant risks or IRB/equivalent review details.

    Guidelines:

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

16. **Declaration of LLM usage**

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

    Answer: [Yes]

    Justification: The core method uses VLM/LLM components for prompt decomposition, structured feedback generation, and differentiable reward supervision, and these uses are described in the paper.

    Guidelines:

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