GOLDEN ANCHOR FOR VIDEO GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Video generation supports digital content creation, with applications in virtual avatars, AI-driven filmmaking, interactive game engines, and world models. State-of-the-art systems achieve controllability primarily through image-to-video (I2V) generation, in which an injected *anchor* image determines the identity, composition, and style of the resulting video. In modern pipelines, this anchor is synthesized by a text-to-image (T2I) generator. This raises a largely overlooked question: *Are images obtained directly from the T2I model truly suitable for current video models?* Our analysis shows that anchor aesthetic and quality scores are weakly correlated with downstream video quality. The structure-and-texture score is negatively correlated with I2V video quality in both the visual and motion dimensions. Fine-tuning the T2I model on an image reward accordingly yields anchors with visibly denser structural and character detail: they score higher on image metrics, yet lower the video reward on every I2V model we test and change downstream motion unpredictably. We therefore introduce **Golden Anchor Learning** (**GAL**), to our knowledge the first framework for learning golden anchors by optimizing a T2I generator with downstream video feedback. GAL evaluates candidate anchors through the videos they induce under a frozen I2V model and converts video rewards into image-level preferences for DPO fine-tuning of the T2I generator. On all four I2V models, GAL raises the VBench average by 2.97%–8.46% over the untuned generator. The rendered clips also score higher under VideoReward-Overall, and their extracted frames under HPSv3 and UniPercept. GAL yields consistent motion gains across all four I2V models, as measured by a 5.62%–38.24% increase in VBench dynamic degree. Although GAL is trained only on HunyuanVideo-1.5 renders, these gains extend to the other three I2V models.

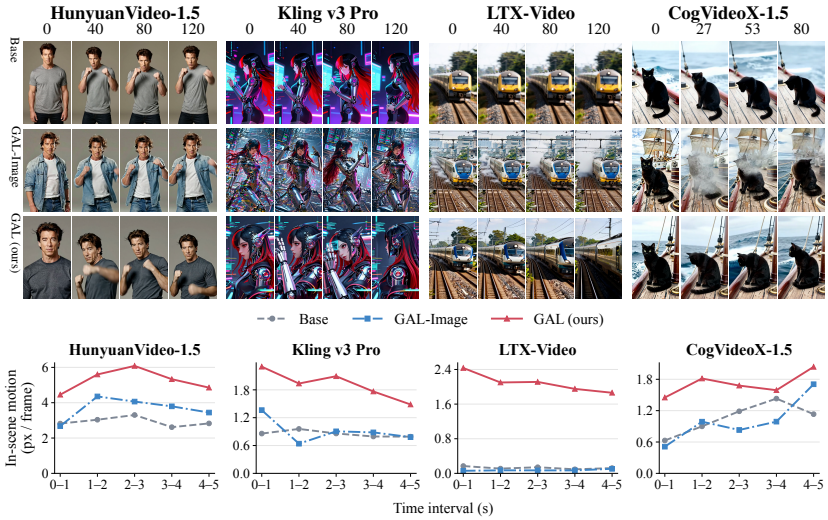


Figure 1: **In the shown examples, the GAL anchor induces more in-scene motion in every one-second interval across all four I2V models.** Top: within each I2V-model block, rows are clips rendered from Base (the anchor produced by the untuned image model), GAL-Image (trained based on image rewards), and our GAL anchor (trained based video rewards), under the same prompt and renderer; four frames span each clip. Bottom: line plots of per-interval motion for the same clips, measured as the mean residual optical-flow magnitude inside each one-second interval (Appendix G).

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

1 INTRODUCTION

Video generation (Kong et al., 2024; HaCohen et al., 2024; Kling Team, 2025) now powers virtual avatars (Lin et al., 2025; Tian et al., 2024), AI films (Polyak et al., 2024; Wu et al., 2025b; Wang et al., 2026), and interactive world models (Bruce et al., 2024; NVIDIA, 2025; Valevski et al., 2025). One practical approach is a text-to-image-to-video pipeline (Girdhar et al., 2025): a text-to-image (T2I) model synthesizes an anchor image (Podell et al., 2024; Esser et al., 2024; Z-Image Team et al., 2025)[1], and an image-to-video (I2V) model animates it (Blattmann et al., 2023; Xing et al., 2024; Yang et al., 2025). The anchor strongly influences the visual content of the resulting clip (Girdhar et al., 2025; Chen et al., 2026). Agentic pipelines inspect and refine keyframes (Li et al., 2024; Wang et al., 2026), and evaluate and select candidate video clips (Shi et al., 2025).

Image-based anchor selection uses image quality as a proxy for downstream video quality. Candidate anchor images are ranked by aesthetics, prompt–image alignment (Guo et al., 2024), or human judgment of the image. These criteria do not account for how the I2V model responds to the anchor. I2V models can over-rely on the conditioning image (Zhao et al., 2024; Ge et al., 2025), and high-frequency detail exposed early in sampling can suppress motion (Choi et al., 2026).

Our experiments show that image rewards do not reliably guide anchor optimization. Across 49,416 (prompt–anchor–video) records, the anchor's aesthetic and quality scores are nearly uncorrelated with video rewards, while its structure/texture score is negatively correlated with the video's visual and motion quality (Figure 2). Fine-tuning the T2I generator on image-based preferences produces sharper anchors but lowers VideoReward-Overall on all four evaluated I2V models (Tables 4 and 8), with motion decreasing on some models and increasing on others (Table 3).

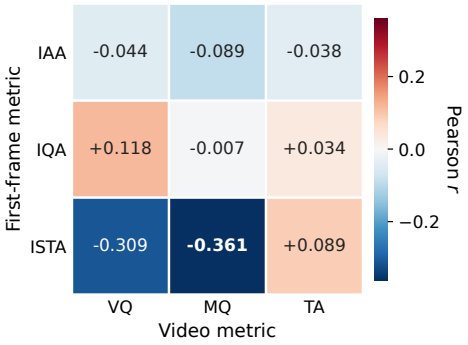


Figure 2: **Mismatch between image scores and video rewards.** Pearson correlation between UniPercept scores of the anchor (IAA: aesthetics; IQA: quality; ISTA: structure and texture) and VideoReward scores of its HunyuanVideo-1.5 render (VQ: visual quality; MQ: motion quality; TA: text alignment) over 49,416 (prompt–anchor–video) records. Aesthetics and quality are nearly uncorrelated with every video axis, and structure/texture is anti-correlated with visual and motion quality.

We therefore make the T2I generator itself *video-aware*: an anchor should be optimized for the video it induces, not for its standalone appearance. We formalize this target through the notion of a *golden anchor*: an anchor's value is determined by the quality of the video generated from it by a frozen I2V model. Golden anchor learning therefore optimizes the T2I generator for this downstream objective (Section 3; cf. Cheng et al., 2025). Our hypothesis is that the motion and quality biases of a video diffusion model can be injected upstream into the image prior of a T2I generator: the anchor is not a static preview but a control surface for downstream dynamics and quality.

Contributions. To our knowledge, this is the first work to improve image-to-video generation by learning a better anchor rather than selecting one (Guo et al., 2024), and the first to formalize anchor generation as a learning problem (Eq. (2)). Concretely:

- **Finding: image quality is misaligned with downstream video quality.** Across 49,416 (prompt–anchor–video) records, the UniPercept structure-and-texture axis is negatively correlated with the visual quality and motion quality axes of VideoReward (Figure 2). This mismatch makes image rewards unreliable objectives for anchor optimization. Anchors chosen by image-side criteria can therefore induce worse videos, so anchor supervision must come from the video itself.
- **Method: Golden Anchor Learning (GAL).** To our knowledge, GAL is the first framework to train a T2I generator using downstream video rewards from a frozen I2V model. It converts video

[1]We use *anchor* rather than *first frame*: in many I2V systems, the conditioning image guides the generation without necessarily being the literal first frame of the output.

scores into image-level preferences for offline DPO (Rafailov et al., 2023; Wallace et al., 2024), requiring no additional rendering or scoring at inference.

- **Result: GAL improves video quality and motion across I2V models.** On all four I2V models, GAL raises the VBench average by 2.97%–8.46% over the untuned generator (Table 1), and the rendered clips score higher under VideoReward-Overall and their frames under HPSv3 and UniPercept (Table 2). Motion gains are consistent across the four I2V models, with dynamic degree rising by 5.62%–38.24%, alongside improvements in imaging and aesthetic quality (Table 1). GAL is trained on HunyuanVideo-1.5 renders alone, yet the gains transfer to LTX-Video, CogVideoX-1.5, and the closed-source Kling v3 Pro.

2 Related Work

Image-anchored video generation pipelines. Many high-quality video-generation workflows follow a T2I→I2V decomposition (Girdhar et al., 2025; Li et al., 2024; Wang et al., 2026): a text-to-image model first synthesizes an anchor image (Esser et al., 2024; Z-Image Team et al., 2025), which a downstream image-to-video model then animates into a clip (Blattmann et al., 2023; Xing et al., 2024; Zhang et al., 2023). Golden Noise learns prompt-conditioned initial noise (Zhou et al., 2025), while Zigzag sampling refines denoising trajectories (Bai et al., 2025) and CoRe^2 improves T2I quality and speed (Shao et al., 2026). Prior work shows how strongly the anchor governs the render: I2V diffusion models over-rely on the conditioning image at large timesteps and lose motion (Zhao et al., 2024; Jeon et al., 2026), early exposure of its high-frequency detail steers sampling toward a static copy of the reference (Choi et al., 2026), and the first frame acts as a memory buffer for visual entities (Chen et al., 2026). IV-Mixed Sampler combines image and video diffusion models during sampling (Shao et al., 2025). Agentic pipelines also incorporate inference-time feedback: CoTriSyGen analyzes generated keyframes and videos to refine prompts and memory, and can trigger keyframe regeneration (Yin et al., 2026). These methods act through image conditioning, rendering dynamics, or inference-time feedback; our method, Golden Anchor Learning (GAL), instead trains the upstream generator to shift the anchor distribution, so the two are complementary. The anchor influences the clip yet is not the final artifact, so its value should be assessed through the video it induces, which we formalize next (Section 3).

Preference learning across image and video. DPO and its variants (Rafailov et al., 2023; Azar et al., 2024; Meng et al., 2024) give an offline alternative to PPO-style preference optimization, and Diffusion-DPO carries the idea to image diffusion (Wallace et al., 2024; Liu et al., 2026a). The reward signal is usually image-domain, however: HPSv2/v3 and UniPercept score images (Wu et al., 2023; Ma et al., 2025; Cao et al., 2026). Optimizing a T2I generator with these rewards can improve anchor appearance, but its effect on downstream video motion is unstable across I2V models. Video-domain rewards such as VideoReward instead score rendered clips (Liu et al., 2025b). Recent work optimizes video generators directly from preferences at the clip, segment, or physical level (Liu et al., 2025c; Wu et al., 2025c; Chen et al., 2025). Closest in structure, VPO tunes an upstream prompt rewriter with preference learning driven by video-level feedback from the rendered clips, and the rewriter generalizes across video generators (Cheng et al., 2025); we instead optimize the anchor image, on which the I2V model over-relies. Our setting is complementary and makes the image-vs-video distinction central: because the anchor is not the deployed artifact, we keep the I2V model frozen and optimize the T2I generator using rendered-video preferences.

3 Problem Setup: The Golden Anchor of an I2V Render

We view a T2I→I2V pipeline as a two-stage conditional generator. Given a prompt p, the upstream image generator π_ψ samples a candidate anchor $x \sim \pi_\psi(\cdot \mid p)$, which then conditions a *frozen* downstream I2V model G to produce a rendered clip $V = G(x, p, z)$, where $z \sim \mathcal{N}(0, 1)$ denotes i.i.d. I2V noise, interpreted elementwise. We call x the *anchor* of the render: unlike standard T2I preference learning, x is not the final output, and its downstream value is assessed through the video it induces. For a fixed I2V model G and video reward R_v, the downstream value of an anchor is

$$O(x, p) = \mathbb{E}_{z\sim\mathcal{N}(0,1)}\left[R_v\big(G(x, p, z)\big)\right], \tag{1}$$

For fixed, frozen (G, R_v) and prompt p, a *golden anchor* is an image with high $O(x, p)$: it tends to induce highly rewarded videos across rendering noise. The *golden anchor objective* shifts the T2I

distribution toward these anchors:

$$\max_{\psi} \ \mathbb{E}_p \mathbb{E}_{x \sim \pi_\psi(\cdot|p)} \left[O(x, p) \right]. \tag{2}$$

Estimating $O(x, p)$ requires rendering and scoring videos; HunyuanVideo-1.5 renders one five-second clip in roughly five minutes on a single RTX 4090. This makes online reinforcement learning (Black et al., 2024; Shao et al., 2024; Liu et al., 2025a; Zheng et al., 2026) expensive in our setting, because collecting fresh rollouts repeatedly invokes the I2V model. We therefore adopt an offline strategy: we pay the video-reward scorer once to build a preference dataset over *sibling anchors*, candidate images sampled for the same prompt, then optimize π_ψ against these fixed preferences with DPO, keeping the expensive I2V renderer out of the training loop (Section 4).

4 GAL: Golden Anchor Learning

GAL first builds a fixed preference dataset $\mathcal{D}_{\text{pref}}$ of tuples (p, x_w, x_l) labeled in video space (Section 4.1), then optimizes the upstream image generator against these preferences while keeping the I2V renderer and video reward frozen (Section 4.2). Algorithm 1 summarizes the procedure.

4.1 Constructing cross-modal preferences

Figure 3 shows how GAL converts downstream video scores into image-level preferences. For each prompt p, we draw $K = 4$ sibling anchors from the initial policy π_{ψ_0} using the reverse-time SDE in Appendix B. We use the untuned Z-Image-Turbo model as the initial policy. Independent Brownian paths make the sampling trajectories stochastic and provide the within-prompt diversity needed for pairwise comparison (Song et al., 2021).

A frozen I2V model renders each anchor once, $V_i = G(x_i, p, z_i)$ (Wu et al., 2025a), and a frozen video-reward model assigns $s_i = R_v(V_i)$ (Liu et al., 2025b). We use the scalar *Overall* score for training and retain the per-axis scores only for analysis. For a higher-scoring anchor x_w and a lower-scoring anchor x_l, we keep the pair and write

$$x_w \succ_p x_l \quad \text{whenever} \quad s_w - s_l \geq m. \tag{3}$$

The margin removes near-ties that are more sensitive to I2V sampling noise. The compared samples remain images, while their labels are determined by rendered videos.

4.2 Preference optimization via DPO

For an anchor image x, let $y_0 = \text{Enc}(x)$ be its clean latent. Draw a shared noising context $c = \{(\tau_j, \epsilon_j)\}_{j=1}^{M} \sim q_c$ from the base scheduler, and let $y_{\tau_j} = \text{NoiseScheduler}(y_0, \epsilon_j, \tau_j)$. Using the flow-matching residual (Lipman et al., 2023; Liu et al., 2023), define

$$\ell_\psi(x, p; c) = \frac{1}{M} \sum_{j=1}^{M} w(\tau_j) \left\| \widehat{u}_\psi(y_{\tau_j}, \tau_j, p) - u_{\tau_j}(y_0, \epsilon_j) \right\|_2^2. \tag{4}$$

Here $\widehat{u}_\psi$ is the trainable model prediction, u_{τ_j} is the target velocity, and ℓ_{ref} is defined identically for the frozen reference model. We reuse the same context c for the winner, the loser, the trainable model, and the reference model.

Adapting DPO and Diffusion-DPO (Rafailov et al., 2023; Wallace et al., 2024; Liu et al., 2025b) to the flow-matching loss above, we define $\Delta_\psi(x, p; c) = \ell_{\text{ref}}(x, p; c) - \ell_\psi(x, p; c)$ and $d_\psi = \Delta_\psi(x_w, p; c) - \Delta_\psi(x_l, p; c)$. The GAL objective is

$$\mathcal{L}_{\text{DPO}}(\psi) = -\mathbb{E}_{(p, x_w, x_l) \sim \mathcal{D}_{\text{pref}},\ c \sim q_c} \left[\log \sigma(\beta d_\psi) \right]. \tag{5}$$

The loss decreases when fine-tuning reduces the flow-matching residual more for the video-preferred anchor than for the rejected anchor. The renderer and reward model are used only to build $\mathcal{D}_{\text{pref}}$.

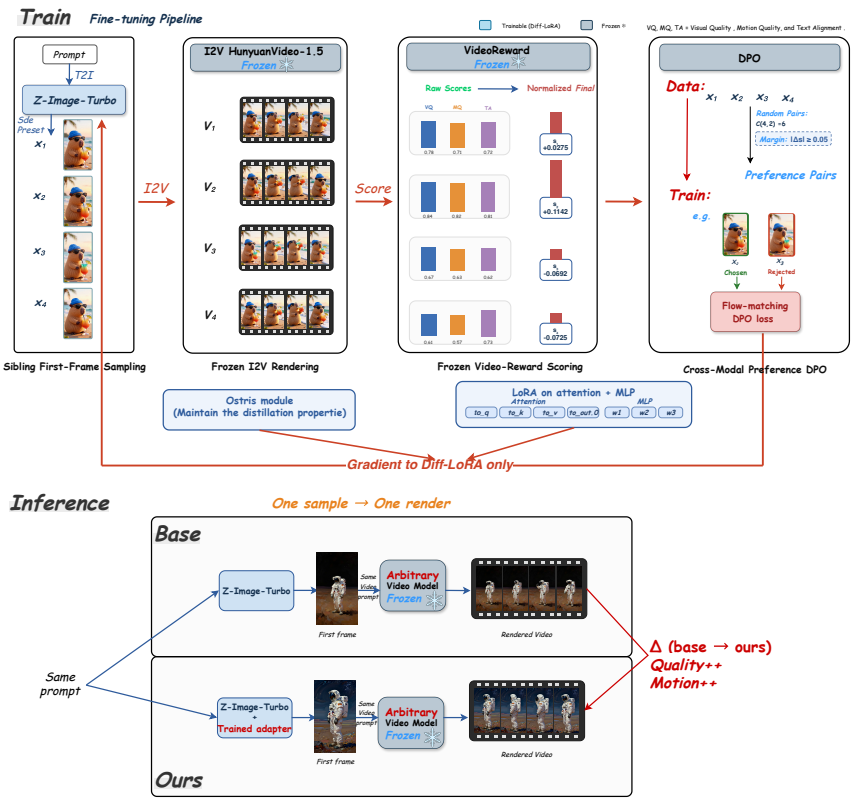


Figure 3: **Overview of GAL. Training (top):** For each prompt, we sample $K = 4$ anchor images from the T2I generator, render each into a video with a frozen I2V model, and score the videos with a frozen video-reward model. We use these scores to construct image-level preference pairs and fine-tune only the upstream T2I generator with cross-modal DPO. Video rendering, scoring, and pair construction are performed offline. **Inference (bottom):** The fine-tuned T2I generator produces anchor images that improve video quality, while the downstream I2V model remains unchanged.

4.3 Few-step implementation

Adapting few-step distilled generators (Salimans & Ho, 2022; Song et al., 2023; Sauer et al., 2024; Z-Image Team et al., 2025; Liu et al., 2026b) can improve image metrics while degrading downstream motion. We train a task LoRA (Hu et al., 2022) within the differential-LoRA scaffold for Z-Image-Turbo (Ostris, 2025; kelseye.xh & Duan, 2025). The frozen preset adapter is attached to the trainable and reference models during training and removed at inference (Appendix C); Eq. (2) targets the deployed generator. Siblings are sampled without adapters; Appendix A discusses the collection/reference mismatch. Removing the scaffold under identical video-reward pairs reverses the motion gain despite image-side improvements (Section 6.5 and Table 3).

5 Analysis of Golden Anchor Learning

We motivate video-derived labels and margin filtering at the objective level. Appendix A gives supporting derivations, and Section 6.3 evaluates the trained system.

Why image and video feedback can disagree. The renderer maps an anchor to an entire clip, so an image change that improves an image score need not improve downstream value $O(x, p)$. For intuition, let $R_{\text{img}}(x, p)$ be an image reward. Even when both objectives are differentiable, the image-improving direction $\nabla_x R_{\text{img}}(x, p)$ need not align with $\nabla_x O(x, p)$: the latter also depends on how the frozen renderer responds to the anchor (Appendix A). This is an objective-level intuition, not the GAL update rule. GAL never computes either gradient; O enters training only through preference labels

Algorithm 1 GAL: Golden Anchor Learning

Require: Prompt set $\mathcal{P}$; pretrained T2I policy π_{ψ_0}; frozen reference π_{ref}; frozen I2V renderer G; frozen video reward R_v; T2I noising-context distribution q_c; sibling count K; margin m; DPO scale β; step size η

Phase 1: Offline preference construction

$\mathcal{D}_{\text{pref}} \leftarrow \varnothing$
for $p \in \mathcal{P}$ **do**
  Sample K sibling anchor images $x_1, \ldots, x_K \sim \pi_{\psi_0}(\cdot \mid p)$ via Eq. (10) with independent Brownian paths
  for $k = 1, \ldots, K$ **do**
    $z_k \sim \mathcal{N}(0, 1)$
    $V_k \leftarrow G(x_k, p, z_k)$
    $s_k \leftarrow R_v(V_k)$
  end for
  for all ordered pairs (i, j) satisfying $s_i - s_j \geq m$ **do**
    $\mathcal{D}_{\text{pref}} \leftarrow \mathcal{D}_{\text{pref}} \cup \{(p, x_i, x_j)\}$
  end for
end for

Phase 2: T2I preference optimization

$\psi \leftarrow \psi_0$
while not converged **do**
  Sample $(p, x_w, x_l) \sim \mathcal{D}_{\text{pref}}$
  Sample a shared T2I noising context $c \sim q_c$
  Evaluate ℓ_ψ and ℓ_{ref} for both anchors under the same c using Eq. (4)
  $d_\psi \leftarrow [\ell_{\text{ref}}(x_w, p; c) - \ell_\psi(x_w, p; c)] - [\ell_{\text{ref}}(x_l, p; c) - \ell_\psi(x_l, p; c)]$
  $\psi \leftarrow \psi - \eta \nabla_\psi [-\log \sigma(\beta d_\psi)]$
end while
return π_ψ

estimated by rendering and scoring candidate anchors. The correlations and GAL-Image control in Section 6.3 provide empirical evidence of image–video misalignment, motivating video-derived labels.

The margin removes uncertain comparisons. A one-render score contains I2V sampling noise, so comparisons with small score gaps are the most vulnerable to order reversal. The margin m in Eq. (3) removes these near-ties before training. Increasing m yields a smaller but more separated preference set (Table 5). We treat the margin as a practical noise-control choice, not as a proof that one render exactly estimates $O(x, p)$. The multi-seed evaluation in Section 6.3 tests whether the learned effect persists under downstream rendering stochasticity.

6 Experiments

The experiments test whether rendered-video feedback produces more effective anchors than image-based feedback. Sections 6.2 and 6.3 present the main comparison across four frozen I2V models, Section 6.4 selects the reward signal within GAL, and Section 6.5 ablates the training scaffold.

6.1 Settings

Variants. All fine-tuned variants share the sibling anchors and the LoRA-DPO recipe of Section 4 and differ only in the signal that labels a pair. Video-supervised variants are named by their rewards: GAL-VideoReward uses VideoReward-Overall on the HunyuanVideo-1.5 render; GAL-HPSv3 and GAL-Unipercept use the corresponding image rewards on frames of that render; GAL-Image uses UniPercept and HPSv3 on the anchor image itself. Base is the untuned generator, and GAL-VideoReward w/o scaffold removes the preset adapter of Section 4.3. GAL-VideoReward is our default and appears as GAL (ours) in Tables 1 and 2. Hyperparameters are listed in Appendix C.

Table 1: Main VBench comparison across four I2V models. GAL (ours), the GAL-VideoReward variant, improves the VBench average, dynamic degree, imaging quality, and aesthetic quality over Base on every model. Parentheses give percentage changes from Base; bold marks the best within each model. Open-source results average three I2V seeds; Kling v3 Pro uses one seed.

Variant	Dynamic Degree	Imaging Quality	Aesthetic Quality	Subject Consistency	Motion Smoothness	Overall Consistency	VBench Average
HunyuanVideo-1.5							
Base	0.8241	0.6586	0.6160	0.9577	**0.9922**	0.2247	0.7122
GAL (ours)	**0.8704** (+5.62%)	**0.7111** (+7.97%)	**0.6426** (+4.32%)	**0.9601** (+0.25%)	0.9909 (−0.13%)	**0.2248** (+0.07%)	**0.7333** (+2.97%)
LTX-Video (transfer)							
Base	0.5509	0.6493	0.5909	0.9557	**0.9922**	0.2537	0.6654
GAL (ours)	**0.6389** (+15.97%)	**0.7076** (+8.99%)	**0.6064** (+2.62%)	**0.9613** (+0.59%)	0.9903 (−0.19%)	**0.2551** (+0.53%)	**0.6933** (+4.18%)
CogVideoX-1.5 (transfer)							
Base	0.7222	0.6121	0.5868	0.9203	**0.9716**	0.2490	0.6770
GAL (ours)	**0.7731** (+7.05%)	**0.6749** (+10.26%)	**0.6016** (+2.52%)	**0.9219** (+0.17%)	0.9633 (−0.86%)	**0.2527** (+1.45%)	**0.6979** (+3.09%)
Kling v3 Pro (closed-source transfer)							
Base	0.6800	0.6721	0.6607	0.9576	**0.9928**	0.2434	0.7011
GAL (ours)	**0.9400** (+38.24%)	**0.7263** (+8.07%)	**0.7018** (+6.22%)	**0.9609** (+0.35%)	0.9888 (−0.41%)	**0.2448** (+0.57%)	**0.7604** (+8.46%)

Table 2: Main reward-model comparison across four I2V models. GAL (ours) improves all three reward scores over Base on every model. VideoReward-Overall scores rendered clips; HPSv3 and UniPercept score their extracted frames. Parentheses give absolute changes from Base.

I2V model	Variant	VideoReward Overall	HPSv3 (video frames)	UniPercept (video frames)
HunyuanVideo-1.5	Base	−0.014	7.0025	51.4375
	GAL (ours)	**+0.131** (+0.145)	**7.6286** (+0.6261)	**55.5926** (+4.1550)
LTX-Video (transfer)	Base	−0.328	6.4788	50.0465
	GAL (ours)	**−0.224** (+0.103)	**7.3407** (+0.8619)	**54.1946** (+4.1481)
CogVideoX-1.5 (transfer)	Base	−0.284	5.1392	46.5880
	GAL (ours)	**−0.129** (+0.155)	**6.1601** (+1.0209)	**51.4183** (+4.8303)
Kling v3 Pro (closed-source transfer)	Base	+0.394	8.2367	59.5410
	GAL (ours)	**+0.909** (+0.515)	**8.9403** (+0.7035)	**62.4870** (+2.9460)

I2V models. HunyuanVideo-1.5 alone supplies labels. LTX-Video, CogVideoX-1.5, and the closed-source Kling v3 Pro never label a pair and test transfer to unseen renderers. We call these four signal-bearing: their VBench dynamic degree keeps motion headroom and responds to the anchor.

Evaluation. Each variant generates one anchor per prompt under a fixed T2I sampling setting. On the open-source renderers we use the official VBench per-dimension prompt sets (Huang et al., 2024) with one fixed video-instruction template, render each anchor under three I2V seeds, and average the VBench scores, to reduce the contribution of I2V sampling noise to the estimated anchor effect. Kling v3 Pro renders a held-out set of 50 prompts once. Reward-model scores (Table 2) are computed on the rendered clips, with VideoReward-Overall on the clip and HPSv3 and UniPercept on its extracted frames. Training, VBench, and Kling prompts are pairwise disjoint (Appendix D).

6.2 Qualitative Comparison

Figure 1 provides a visual check of the motion claim. For each I2V model, it shows clips rendered from Base, GAL-Image, and GAL-VideoReward under the same prompt and renderer. In the shown examples, GAL-VideoReward produces larger frame-to-frame changes, and its per-interval in-scene motion is the highest of the three variants in all 20 one-second intervals across the four I2V models. This pattern matches the dynamic-degree gains in Table 1.

6.3 Main Comparison across I2V Models

Image metrics misalign with the video reward. Figure 2 gives the full correlation matrix behind the finding in Section 1: no aesthetics or quality correlation exceeds 0.118 in magnitude, while the

Table 3: Reward-signal ablation within GAL, evaluated with VBench across four I2V models. Parentheses give percentage changes from Base; bold marks the best within each model. GAL-VideoReward is the default variant.

Variant	Dynamic Degree	Imaging Quality	Aesthetic Quality	Subject Consistency	Motion Smoothness	Overall Consistency	VBench Average
HunyuanVideo-1.5							
Base	0.8241	0.6586	0.6160	0.9577	**0.9922**	0.2247	0.7122
GAL-Image	0.7917(−3.93%)	**0.7302**(+10.87%)	0.6453(+4.76%)	0.9568(−0.09%)	0.9902(−0.20%)	0.2230(−0.76%)	0.7229(+1.50%)
GAL-HPSv3	**0.8796**(+6.74%)	0.7035(+6.82%)	0.6310(+2.43%)	0.9529(−0.50%)	0.9882(−0.41%)	**0.2248**(+0.03%)	0.7300(+2.50%)
GAL-Unipercept	0.8380(+1.69%)	0.7038(+6.86%)	**0.6638**(+7.75%)	0.9564(−0.13%)	0.9911(−0.11%)	0.2223(−1.07%)	0.7292(+2.39%)
GAL-VideoReward	0.8704(+5.62%)	0.7111(+7.97%)	0.6426(+4.32%)	**0.9601**(+0.25%)	0.9909(−0.13%)	**0.2248**(+0.07%)	**0.7333**(+2.97%)
↪ w/o scaffold	0.8056(−2.25%)	0.7147(+8.52%)	0.6413(+4.11%)	0.9599(+0.23%)	0.9911(−0.12%)	0.2241(−0.24%)	0.7228(+1.48%)
LTX-Video (transfer)							
Base	0.5509	0.6493	0.5909	0.9557	**0.9922**	0.2537	0.6654
GAL-Image	0.5278(−4.20%)	**0.7287**(+12.23%)	0.6211(+5.13%)	**0.9636**(+0.83%)	0.9899(−0.23%)	0.2565(+1.11%)	0.6813(+2.38%)
GAL-HPSv3	0.5278(−4.20%)	0.7136(+9.90%)	0.6148(+4.06%)	0.9627(+0.73%)	0.9897(−0.25%)	0.2518(−0.75%)	0.6767(+1.70%)
GAL-Unipercept	0.5972(+8.40%)	0.7052(+8.62%)	**0.6299**(+6.60%)	0.9575(+0.19%)	0.9903(−0.19%)	**0.2575**(+1.51%)	0.6896(+3.63%)
GAL-VideoReward	**0.6389**(+15.97%)	0.7076(+8.99%)	0.6064(+2.62%)	0.9613(+0.59%)	0.9903(−0.19%)	0.2551(+0.53%)	**0.6933**(+4.18%)
CogVideoX-1.5 (transfer)							
Base	0.7222	0.6121	0.5868	0.9203	**0.9716**	0.2490	0.6770
GAL-Image	0.7546(+4.49%)	**0.6942**(+13.41%)	0.6088(+3.75%)	0.9060(−1.55%)	0.9529(−1.92%)	0.2501(+0.44%)	0.6944(+2.57%)
GAL-HPSv3	0.7639(+5.77%)	0.6772(+10.64%)	0.5967(+1.69%)	0.9089(−1.24%)	0.9470(−2.53%)	0.2493(+0.12%)	0.6905(+1.99%)
GAL-Unipercept	0.7639(+5.77%)	0.6776(+10.70%)	**0.6213**(+5.88%)	0.9162(−0.45%)	0.9601(−1.18%)	**0.2542**(+2.09%)	**0.6989**(+3.23%)
GAL-VideoReward	**0.7731**(+7.05%)	0.6749(+10.26%)	0.6016(+2.52%)	**0.9219**(+0.17%)	0.9633(−0.86%)	0.2527(+1.45%)	0.6979(+3.09%)
Kling v3 Pro (closed-source transfer)							
Base	0.6800	0.6721	0.6607	0.9576	**0.9928**	0.2434	0.7011
GAL-Image	0.9000(+32.35%)	**0.7470**(+11.15%)	0.7016(+6.19%)	0.9424(−1.59%)	0.9861(−0.68%)	0.2329(−4.32%)	0.7517(+7.21%)
GAL-HPSv3	0.8400(+23.53%)	0.7236(+7.68%)	0.6709(+1.54%)	0.9528(−0.51%)	0.9865(−0.64%)	0.2352(−3.37%)	0.7348(+4.81%)
GAL-Unipercept	0.8200(+20.59%)	0.7223(+7.48%)	**0.7342**(+11.12%)	0.9540(−0.38%)	0.9905(−0.23%)	0.2428(−0.27%)	0.7440(+6.11%)
GAL-VideoReward	**0.9400**(+38.24%)	0.7263(+8.07%)	0.7018(+6.22%)	**0.9609**(+0.35%)	0.9888(−0.41%)	**0.2448**(+0.57%)	**0.7604**(+8.46%)

Table 4: Reward-signal ablation within GAL, evaluated by video and frame rewards on HunyuanVideo-1.5. Among the compared variants, GAL-VideoReward alone improves all three scores over Base on HunyuanVideo-1.5 and all three transfer models (Table 8 in Appendix H). We therefore use it as the default variant. Metrics and parentheses follow Table 2.

Variant	VideoReward Overall	HPSv3 (video frames)	UniPercept (video frames)
Base	−0.014	7.0025	51.4375
GAL-Image	−0.531 (−0.517)	7.9723 (+0.9698)	57.2618 (+5.8243)
GAL-HPSv3	−0.493 (−0.478)	7.9509 (+0.9484)	55.7763 (+4.3388)
GAL-Unipercept	−0.171 (−0.157)	**7.9841** (+0.9816)	**58.6379** (+7.2004)
GAL-VideoReward	**+0.131** (+0.145)	7.6286 (+0.6261)	55.5926 (+4.1550)

structure-and-texture axis is anti-correlated with both visual quality and motion quality. Standalone image metrics therefore do not reliably rank anchors by downstream video value.

GAL-VideoReward improves video quality and motion across I2V models. Across all four I2V models, GAL-VideoReward raises the VBench average by 2.97%–8.46% over Base (Table 1). Compared with Base, its rendered clips score higher under VideoReward-Overall, and their frames score higher under HPSv3 and UniPercept (Table 2). Dynamic degree increases by 5.62% on HunyuanVideo-1.5, 15.97% on LTX-Video, 7.05% on CogVideoX-1.5, and 38.24% on Kling v3 Pro (Table 1). Imaging quality rises by 7.97%–10.26% and aesthetic quality by 2.52%–6.22% across the four I2V models. Only HunyuanVideo-1.5 supplies labels; gains on unseen renderers show transfer.

6.4 Reward-Signal Ablation within GAL

Among the tested reward variants, GAL-VideoReward attains the highest VBench average on three I2V models and the second highest on CogVideoX-1.5 (Table 3). It is also the only variant whose rendered clips score above Base under VideoReward-Overall and whose extracted frames score above Base under both HPSv3 and UniPercept on all four I2V models (Tables 4 and 8). For motion, GAL-VideoReward achieves the highest dynamic degree on three models and the second highest on HunyuanVideo-1.5 (Table 3). These consistent improvements in overall performance and motion

support our choice of VideoReward-Overall for preference construction and GAL-VideoReward as the default variant.

Image supervision depends on the I2V model. Compared with GAL-Image, GAL-VideoReward achieves higher VideoReward-Overall and dynamic degree on all four I2V models (Tables 3, 4 and 8). GAL-Image yields higher imaging quality on all four models and higher aesthetic quality on three, but scores below Base under VideoReward-Overall on every model. Its dynamic-degree change ranges from -4.20% on LTX-Video to $+32.35\%$ on Kling v3 Pro. GAL-VideoReward improves imaging quality and dynamic degree over Base; image-quality gains need not entail motion loss.

6.5 Scaffold Ablation

Removing the differential-LoRA scaffold reverses the motion gain of GAL-VideoReward. With preference pairs, architecture, hyperparameters, and checkpoint fixed, dropping the scaffold changes HunyuanVideo-1.5 dynamic degree from $+5.62\%$ to -2.25%, while imaging and aesthetic quality still rise above Base (Table 3). The unscaffolded model scores higher under VideoReward-Overall ($+0.289$ against $+0.145$ above Base) yet loses the motion gain, so a higher training-reward score alone does not secure the motion benefit. The scaffold therefore acts as a stability guardrail for the motion benefit rather than a generic image-quality booster.

6.6 Discussion

Limitations. First, we use only HunyuanVideo-1.5 for preference construction to test whether GAL transfers to unseen I2V models. The consistent gains on all three unseen backends demonstrate this transferability (Tables 1 and 2). However, HunyuanVideo-1.5 exhibits lower within-prompt variance in the binary dynamic-degree readout across anchor seeds than LTX-Video and CogVideoX-1.5 (Appendix E). Constructing preferences with a backend whose dynamic-degree readout is more sensitive to anchor changes may yield larger motion gains. Combining preferences from multiple backends could also further improve anchor learning. Second, GAL depends strongly on the choice and reliability of the reward model used for preference construction. The limited choice of video reward models restricts supervision, and different reward signals can favor conflicting qualities. Small motion-smoothness regressions also remain under our default reward (Table 1). Preference objectives that account for reward disagreements and temporal stability could improve anchor learning. As video reward models improve, we expect GAL to learn more effective anchors through more reliable preference supervision.

Implications. Anchor learning offers a way to align image and video generation without modifying the downstream model. Video feedback shapes the upstream image prior during training, rather than requiring additional selection or correction at inference. This makes anchor learning a separate optimization component that could complement improvements to I2V architectures and samplers. The observed transfer to unseen I2V models further suggests that some video-effective anchor properties are useful beyond the I2V model used to construct training preferences.

7 Conclusion

This paper investigated which images make effective anchors for a downstream video generator and introduced the notion of a golden anchor: an image valued by the video it induces rather than by its standalone appearance. We instantiate this notion as GAL (Golden Anchor Learning). GAL is a cross-modal DPO method that renders sibling anchors through a frozen I2V model, scores them with a frozen video reward, and optimizes the upstream T2I LoRA alone against the resulting preferences. Because all rendering and scoring happen offline during data construction, deployment stays one T2I sample followed by one I2V render. Across four frozen I2V models, GAL raises the VBench average above Base. Its rendered clips also score higher than those from Base under VideoReward-Overall, and their extracted frames score higher under HPSv3 and UniPercept. These gains are accompanied by consistent improvements in dynamic motion, imaging quality, and aesthetic quality. These results show that the motion and quality biases of a frozen video model can guide the upstream T2I generator toward anchors that improve I2V video quality and motion.

AI USE STATEMENT

We used large language models to draft and polish the writing of this manuscript, to check the derivations in the appendix, and to search for related literature. They were not used to design, implement, or run experiments, to analyze data, or to interpret results. All AI-assisted text and derivations were reviewed by the authors, and we take responsibility for the final content of this work.

REPRODUCIBILITY STATEMENT

We provide training hyperparameters and implementation details in Appendix C, and evaluation settings and prompt protocols in Section 6.1 and Appendix D.

REFERENCES

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

Lichen Bai, Shitong Shao, Zikai Zhou, Zipeng Qi, Zhiqiang Xu, Haoyi Xiong, and Zeke Xie. Zigzag diffusion sampling: Diffusion models can self-improve via self-reflection. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=MKvQH1ekeY.

Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning. In *International Conference on Learning Representations*, volume 2024, pp. 4965–4987, 2024.

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

Jake Bruce, Michael D. Dennis, Ashley Edwards, Jack Parker-Holder, Yuge Shi, Edward Hughes, Matthew Lai, Aditi Mavalankar, Richie Steigerwald, Chris Apps, et al. Genie: Generative interactive environments. In *International Conference on Machine Learning*, 2024.

Shuo Cao, Jiayang Li, Xiaohui Li, Yuandong Pu, Kaiwen Zhu, Yuanting Gao, Siqi Luo, Yi Xin, Qi Qin, Yu Zhou, Xiangyu Chen, Wenlong Zhang, Bin Fu, Yu Qiao, and Yihao Liu. UniPercept: Towards unified perceptual-level image understanding across aesthetics, quality, structure, and texture. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=8VVbElV06v.

Harold Haodong Chen, Haojian Huang, Qifeng Chen, Harry Yang, and Ser Nam Lim. Hierarchical fine-grained preference optimization for physically plausible video generation. *Advances in Neural Information Processing Systems*, 38:133919–133951, 2025.

Jingxi Chen, Zongxia Li, Zhichao Liu, Guangyao Shi, Xiyang Wu, Fuxiao Liu, Cornelia Fermüller, Brandon Y. Feng, and Yiannis Aloimonos. First frame is the place to go for video content customization. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026.

Jiale Cheng, Ruiliang Lyu, Xiaotao Gu, Xiao Liu, Jiazheng Xu, Yida Lu, Jiayan Teng, Zhuoyi Yang, Yuxiao Dong, Jie Tang, Hongning Wang, and Minlie Huang. VPO: Aligning text-to-video generation models with prompt optimization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025.

June Suk Choi, Kyungmin Lee, Sihyun Yu, Yisol Choi, Jinwoo Shin, and Kimin Lee. Improving motion in image-to-video models via adaptive low-pass guidance. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *International Conference on Machine Learning*, 2024.

Gunnar Farnebäck. Two-frame motion estimation based on polynomial expansion. In *Scandinavian Conference on Image Analysis (SCIA)*, pp. 363–370, 2003.

Yunyang Ge, Xinhua Cheng, Chengshu Zhao, Xianyi He, Shenghai Yuan, Bin Lin, Bin Zhu, and Li Yuan. FlashI2V: Fourier-guided latent shifting prevents conditional image leakage in image-to-video generation. *arXiv preprint arXiv:2509.25187*, 2025.

Rohit Girdhar, Mannat Singh, Andrew Brown, Quentin Duval, Samaneh Azadi, Sai Saketh Rambhatla, Akbar Shah, Xi Yin, Devi Parikh, and Ishan Misra. Factorizing text-to-video generation by explicit image conditioning. In Aleš Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol (eds.), *Computer Vision – ECCV 2024*, pp. 205–224, Cham, 2025. Springer Nature Switzerland. ISBN 978-3-031-73033-7.

Xiefan Guo, Jinlin Liu, Miaomiao Cui, Liefeng Bo, and Di Huang. I4VGen: Image as free stepping stone for text-to-video generation. *arXiv preprint arXiv:2406.02230*, 2024.

Yoav HaCohen, Nisan Chiprut, Benny Brazowski, Daniel Shalem, Dudu Moshe, Eitan Richardson, Eran Levin, Guy Shiran, Nir Zabari, Ori Gordon, et al. LTX-Video: Realtime video latent diffusion. *arXiv preprint arXiv:2501.00103*, 2024.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*, 2022.

Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, et al. VBench: Comprehensive benchmark suite for video generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21807–21818, 2024.

Wooseok Jeon, Seungho Park, Seunghyun Shin, Sangeyl Lee, Hyeonho Jeong, and Hae-Gon Jeon. Rebalancing reference frame dominance to improve motion in image-to-video models. *arXiv preprint arXiv:2605.19398*, 2026.

kelseye.xh and Zhongjie Duan. Training strategies of Z-Image-Turbo, 2025. URL `https://huggingface.co/blog/kelseye/training-strategies-of-z-image-turbo`. Hugging Face Blog. Accessed: 2026-07-05.

Kling Team. Kling-Omni technical report. *arXiv preprint arXiv:2512.16776*, 2025.

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

Yunxin Li, Haoyuan Shi, Baotian Hu, Longyue Wang, Jiashun Zhu, Jinyi Xu, Zhen Zhao, and Min Zhang. Anim-Director: A large multimodal model powered agent for controllable animation video generation. In *SIGGRAPH Asia 2024 Conference Papers*, pp. 1–11, 2024.

Gaojie Lin, Jianwen Jiang, Jiaqi Yang, Zerong Zheng, Chao Liang, Yuan Zhang, and Jingtuo Liu. OmniHuman-1: Rethinking the scaling-up of one-stage conditioned human animation models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 13847–13858, October 2025.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations*, 2023. URL `https://openreview.net/forum?id=PqvMRDCJT9t`.

Buhua Liu, Shitong Shao, Bao Li, Lichen Bai, Zhiqiang Xu, Haoyi Xiong, James T. Kwok, Sumi Helal, and Zeke Xie. Alignment of diffusion models: Fundamentals, challenges, and future. *ACM Comput. Surv.*, 58(9):244:1–244:37, March 2026a. ISSN 0360-0300. doi: 10.1145/3796982. URL `https://doi.org/10.1145/3796982`.

Dongyang Liu, Peng Gao, David Liu, Ruoyi Du, Zhen Li, Qilong Wu, Xin Jin, Sihan Cao, Shifeng Zhang, Steven HOI, and Hongsheng Li. Decoupled DMD: CFG augmentation as the spear, distribution matching as the shield. In *The Fourteenth International Conference on Learning Representations*, 2026b. URL https://openreview.net/forum?id=jBztvOiCKE.

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. *Advances in neural information processing systems*, 38:40783–40818, 2025a.

Jie Liu, Gongye Liu, Jiajun Liang, Ziyang Yuan, Xiaokun Liu, Mingwu Zheng, Xiele Wu, Qiulin Wang, 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. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 82155–82192. Curran Associates, Inc., 2025b. doi: 10.52202/085713-2750. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/76227feb18ea0ee40bd15cf02c33e18e-Paper-Conference.pdf.

Runtao Liu, Haoyu Wu, Ziqiang Zheng, Chen Wei, Yingqing He, Renjie Pi, and Qifeng Chen. VideoDPO: Omni-preference alignment for video diffusion generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 8009–8019, 2025c.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *International Conference on Learning Representations (ICLR)*, 2023.

Yuhang Ma, Xiaoshi Wu, Keqiang Sun, and Hongsheng Li. HPSv3: Towards wide-spectrum human preference score. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15086–15095, 2025.

Yu Meng, Mengzhou Xia, and Danqi Chen. SimPO: Simple preference optimization with a reference-free reward. *Advances in Neural Information Processing Systems*, 37:124198–124235, 2024.

NVIDIA. Cosmos world foundation model platform for physical AI. *arXiv preprint arXiv:2501.03575*, 2025.

Ostris. Z-Image-Turbo training adapter, 2025. URL https://huggingface.co/ostris/zimage_turbo_training_adapter. Hugging Face model card.

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. SDXL: Improving latent diffusion models for high-resolution image synthesis. In *International Conference on Learning Representations*, volume 2024, pp. 1862–1874, 2024.

Adam Polyak, Amit Zohar, Andrew Brown, Andros Tjandra, Animesh Sinha, Ann Lee, Apoorv Vyas, Bowen Shi, Chih-Yao Ma, Ching-Yao Chuang, et al. Movie Gen: A cast of media foundation models. *arXiv preprint arXiv:2410.13720*, 2024.

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.

Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *International Conference on Learning Representations (ICLR)*, 2022.

Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation. In *European Conference on Computer Vision*, pp. 87–103. Springer, 2024.

Shitong Shao, Zikai Zhou, Lichen Bai, Haoyi Xiong, and Zeke Xie. IV-mixed sampler: Leveraging image diffusion models for enhanced video synthesis. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=ImpeMDJfVL.

Shitong Shao, Zikai Zhou, Dian Xie, Yuetong Fang, Tian Ye, Lichen Bai, Bo Han, and Zeke Xie. Improved and accelerated text-to-image generation with collect, reflect, and refine. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2026.

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.

Haoyuan Shi, Yunxin Li, Xinyu Chen, Longyue Wang, Baotian Hu, and Min Zhang. AniMaker: Multi-agent animated storytelling with MCTS-driven clip generation. In *Proceedings of the SIGGRAPH Asia 2025 Conference Papers*, pp. 1–11, 2025.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations (ICLR)*, 2021.

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. In *International Conference on Machine Learning (ICML)*, 2023.

Linrui Tian, Qi Wang, Bang Zhang, and Liefeng Bo. EMO: Emote portrait alive – generating expressive portrait videos with Audio2Video diffusion model under weak conditions. In *European Conference on Computer Vision*, 2024.

Dani Valevski, Yaniv Leviathan, Moab Arar, and Shlomi Fruchter. Diffusion models are real-time game engines. In *International Conference on Learning Representations*, volume 2025, pp. 73754–73776, 2025.

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*, pp. 8228–8238, 2024.

Wan Team, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Qian Wang, Ziqi Huang, Ruoxi Jia, Paul Debevec, and Ning Yu. MAViS: A multi-agent framework for long-sequence video storytelling. In Vera Demberg, Kentaro Inui, and Lluís Marquez (eds.), *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2273–2295, Rabat, Morocco, March 2026. Association for Computational Linguistics. ISBN 979-8-89176-380-7. doi: 10.18653/v1/2026.eacl-long.101. URL https://aclanthology.org/2026.eacl-long.101/.

Bing Wu, Chang Zou, Changlin Li, Duojun Huang, Fang Yang, Hao Tan, Jack Peng, Jianbing Wu, Jiangfeng Xiong, Jie Jiang, et al. HunyuanVideo 1.5 technical report. *arXiv preprint arXiv:2511.18870*, 2025a.

Weijia Wu, Zeyu Zhu, and Mike Zheng Shou. Automated movie generation via multi-agent CoT planning. *arXiv preprint arXiv:2503.07314*, 2025b.

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

Yujia Wu, Yiming Shi, Jiwei Wei, Chengwei Sun, Yang Yang, and Heng Tao Shen. DiffLoRA: Generating personalized low-rank adaptation weights with diffusion. *arXiv preprint arXiv:2408.06740*, 2024.

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. *Advances in Neural Information Processing Systems*, 38:171632–171668, 2025c.

Jinbo Xing, Menghan Xia, Yong Zhang, Haoxin Chen, Wangbo Yu, Hanyuan Liu, Gongye Liu, Xintao Wang, Ying Shan, and Tien-Tsin Wong. DynamiCrafter: Animating open-domain images with video diffusion priors. In *European Conference on Computer Vision*, pp. 399–417. Springer, 2024.

Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. CogVideoX: Text-to-video diffusion models with an expert transformer. In *International Conference on Learning Representations*, volume 2025, pp. 83048–83077, 2025.

Xinlei Yin, Xiulian Peng, Xiao Li, Zhiwei Xiong, and Yan Lu. Closed-loop triplet synergistic generation for long-form video. *arXiv preprint arXiv:2606.16184*, 2026.

Z-Image Team, Huanqia Cai, Sihan Cao, Ruoyi Du, Peng Gao, Aiming Hao, Steven Hoi, Zhaohui Hou, Shijie Huang, Dengyang Jiang, Yuming Jiang, Xin Jin, Liangchen Li, Zhen Li, Zhong-Yu Li, David Liu, Dongyang Liu, Qilong Wu, Feng Yu, Zechao Zhan, Chi Zhang, Shifeng Zhang, Ruikai Zhou, and Shilin Zhou. Z-Image: An efficient image generation foundation model with single-stream diffusion transformer. *arXiv preprint arXiv:2511.22699*, 2025.

Shiwei Zhang, Jiayu Wang, Yingya Zhang, Kang Zhao, Hangjie Yuan, Zhiwu Qin, Xiang Wang, Deli Zhao, and Jingren Zhou. I2VGen-XL: High-quality image-to-video synthesis via cascaded diffusion models. *arXiv preprint arXiv:2311.04145*, 2023.

Min Zhao, Hongzhou Zhu, Chendong Xiang, Kaiwen Zheng, Chongxuan Li, and Jun Zhu. Identifying and solving conditional image leakage in image-to-video diffusion model. In *Advances in Neural Information Processing Systems*, 2024.

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. In *International Conference on Learning Representations (ICLR)*, 2026.

Zikai Zhou, Shitong Shao, Lichen Bai, Shufei Zhang, Zhiqiang Xu, Bo Han, and Zeke Xie. Golden noise for diffusion models: A learning framework. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 17688–17697, 2025. URL https://openaccess.thecvf.com/content/ICCV2025/html/Zhou_Golden_Noise_for_Diffusion_Models_A_Learning_Framework_ICCV_2025_paper.html.

A Objective-Level Derivations

Pullback gradient. Assume that $G(\cdot, p, z)$ and R_v are differentiable and that $\nabla_x R_v(G(x,p,z))$ is dominated by an integrable function of z in a neighborhood of x. For fixed z, the chain rule gives

$$\nabla_x R_v(G(x,p,z)) = J_x G(x,p,z)^\top \nabla_V R_v(G(x,p,z))\,. \tag{6}$$

The domination assumption permits differentiation under the expectation in Eq. (1), giving

$$\nabla_x O(x,p) = \mathbb{E}_{z\sim\mathcal{N}(0,1)}\left[J_x G(x,p,z)^\top \nabla_V R_v(G(x,p,z))\right], \tag{7}$$

where $J_x G$ is the renderer's Jacobian with respect to the anchor. This identity explains how downstream sensitivity can differ from image-reward sensitivity. It is not evaluated by GAL, which uses only the offline preference labels.

Idealized population interpretation of DPO. This paragraph records an objective-level interpretation, not a convergence claim for the implemented optimizer. Suppose pairwise preferences follow $\Pr(x \succ x' \mid p) = \sigma((O(x,p) - O(x',p))/\lambda)$ with preference-noise scale $\lambda > 0$, and let $h(x,p) = \log(\pi(x \mid p)/\pi_{\mathrm{ref}}(x \mid p))$ be represented exactly. For each pair, the population binary cross-entropy is minimized when $\beta(h(x,p) - h(x',p)) = (O(x,p) - O(x',p))/\lambda$. Full support of the comparison distribution over the reference policy's support therefore makes $\beta h(x,p) - O(x,p)/\lambda$ constant in x. Normalizing the policy yields the standard DPO population target (Rafailov et al., 2023)

$$\pi^\star(x \mid p) = \frac{\pi_{\mathrm{ref}}(x \mid p)\exp(O(x,p)/(\beta\lambda))}{\int \pi_{\mathrm{ref}}(x' \mid p)\exp(O(x',p)/(\beta\lambda))\,\mathrm{d}x'}. \tag{8}$$

The comparison pairs need not be sampled from π_{ref}, but their distribution must have the full pair support assumed above. The assumptions above exclude finite and margin-filtered data, deterministic preference labels, diffusion-loss surrogates, restricted LoRA parameterization, adapter removal at inference, and optimization error. Equation (8) is therefore used only to interpret the direction of the objective.

Effect of the preference margin. Fix two anchors with $O(x_w,p) < O(x_l,p)$; here w designates the anchor that would be incorrectly retained as the winner. Let their one-render scores be $s_k = O(x_k,p) + \varepsilon_k$ for $k \in \{w,l\}$, with independent, mean-zero, ν^2-sub-Gaussian noise. Incorrectly retaining this pair at margin m requires $\varepsilon_w - \varepsilon_l \geq m + O(x_l,p) - O(x_w,p)$. The noise difference is $2\nu^2$-sub-Gaussian, so the standard tail bound gives

$$\Pr(s_w - s_l \geq m) \leq \exp\left(-\frac{[m + O(x_l,p) - O(x_w,p)]^2}{4\nu^2}\right). \tag{9}$$

Under these assumptions, a larger margin suppresses noise-induced inversions at the cost of fewer retained pairs. The bound does not show that one render recovers $O(x,p)$, so the paper treats the margin as a filtering heuristic and evaluates robustness with independent rendering seeds.

B Sibling Anchor Sampling via a Marginal-Preserving Reverse-Time SDE

The few-step Z-Image-Turbo sampler integrates a probability-flow ODE whose trajectory is deterministic given the initial latent. To introduce trajectory stochasticity beyond the initial latent draw, we sample sibling anchors (Section 4.1) with the reverse-time SDE associated with that ODE,

$$\mathrm{d}y_\tau = \left[\, f(y_\tau,\tau) - g(\tau)^2\,\nabla_y \log \rho_\tau(y_\tau \mid p)\,\right]\mathrm{d}\tau + g(\tau)\,\mathrm{d}\bar{W}_\tau, \tag{10}$$

where y_τ is the T2I latent state at diffusion time τ, $\rho_\tau(\cdot \mid p)$ its prompt-conditioned density, and f and g are the drift and diffusion coefficients of the corresponding forward process, $\bar{W}_\tau$ is a reverse-time Brownian motion, and $\nabla_y \log \rho_\tau$ is the score (Song et al., 2021). Because this SDE shares its time-marginals $\{\rho_\tau\}$ with the probability-flow ODE, decoding each terminal latent with the frozen VAE gives a sibling x_k drawn from an approximation of $\pi_{\psi_0}(\cdot \mid p)$, while its Brownian term supplies the within-prompt diversity needed for informative preference pairs. In the data-construction pipeline we draw $K{=}4$ such siblings per prompt under fixed SDE sampler settings (Appendix C).

C HYPERPARAMETERS, COMPUTE BUDGET, AND ADDITIONAL DETAILS

Per-variant training configurations. The four fine-tuned reward configurations reported in the main comparison (GAL-VideoReward, GAL-HPSv3, GAL-Unipercept, GAL-Image), a fifth variant GAL-Unipercept+HPSv3 (both rewards evaluated on extracted video frames, trained identically but omitted from the main tables because it underperformed the other reward signals), and the GAL-VideoReward w/o scaffold ablation (which reuses the VideoReward reward but drops the differential-LoRA scaffold) were all trained as offline DPO LoRA fine-tunes of the same Z-Image-Turbo base checkpoint, with the same trainable LoRA configuration (rank $r = 64$, $\alpha = 64$, dropout 0, target modules {`to_q`, `to_k`, `to_v`, `to_out.0`, `w1`, `w2`, `w3`} on every transformer layer, $158{,}760{,}960$ trainable params out of $6{,}313{,}669{,}696$ total), the same DPO objective ($\beta = 30$, DPO-side clipping $[-5, 5]$, noise-level weighting $w(\tau)$), the same optimizer / scheduler ($\eta = 1 \times 10^{-4}$, weight decay 0, cosine schedule with 1000 warmup steps, max grad norm 1.0), the same dtype and parallelism (bfloat16, FSDP2, gradient checkpointing), and the same image / sequence size (1024 / 512). All evaluations use the step-9000 checkpoint. The training seed is $20\,260\,430$ on every run (a single training seed for the T2I DPO run; the VBench 3-seed averaging is a separate I2V-side evaluation protocol that re-renders each anchor under three downstream I2V rendering seeds, Table 1).

In Eq. (5), $\sigma(a) = 1/(1 + e^{-a})$ is the logistic sigmoid and $\beta > 0$ scales the preference logit.

DPO noising context. In Eq. (4), Enc is the frozen VAE encoder evaluated at its posterior mode, and M is the number of timestep–noise draws in each context. Each preference tuple is evaluated under a single shared noising context $c = (\{\tau_j\}_{j=1}^{M}, \{\epsilon_j\}_{j=1}^{M})$ (Eq. (4)). Concretely, the noisy latent is produced by the base Z-Image-Turbo flow-matching scheduler, $y_{\tau_j} = \mathrm{NoiseScheduler}(y_0, \epsilon_j, \tau_j)$, which realizes the affine map $y_{\tau_j} = a_{\tau_j} y_0 + b_{\tau_j} \epsilon_j$ with the base scheduler's coefficients (a_{τ_j}, b_{τ_j}); the timesteps $\{\tau_j\}_{j=1}^{M}$ are drawn from the same training-time timestep distribution as the base model, the noise draws are standard Gaussian ($\epsilon_j \sim \mathcal{N}(0, I)$), and $w(\tau)$ reuses the base flow-matching timestep weighting, so no separate reward-side schedule is introduced. The timestep set and noise seeds are sampled once, reused for both the winning and rejected images, and reused again when evaluating the trainable and frozen reference generators. Because the image latents are deterministic VAE-mode latents and the winner and loser share the same latent shape, replaying the same noise seeds yields elementwise-identical noise tensors on the two sides of the pair; an explicit assertion checks that the chosen and rejected noise seeds match. All four losses $\ell_\psi(x_w, p; c)$, $\ell_{\mathrm{ref}}(x_w, p; c)$, $\ell_\psi(x_l, p; c)$, and $\ell_{\mathrm{ref}}(x_l, p; c)$ are therefore computed under the same c. This matched-context implementation reduces variance in the DPO difference while preserving the standard stochastic flow-matching estimate.

The five reward configurations (and the GAL-VideoReward w/o scaffold ablation) share an underlying $74{,}124$-row pair pool (the same `all_pairs_drop_ties.jsonl` corpus generated from $49{,}416$ candidate records, i.e. $12{,}354$ prompts from the training prompt pool described in Appendix D $\times$ 4 candidates sampled under the SDE sampler settings, disjoint from both the VBench and Kling evaluation prompts), but each variant scores the pairs with its own reward signal (GAL-VideoReward: VideoReward-Overall on the rendered clip; GAL-HPSv3: per-frame mean HPSv3 over the clip; GAL-Unipercept: UniPercept on the frames; GAL-Unipercept+HPSv3: UniPercept + HPSv3 on the frames; GAL-Image: UniPercept + HPSv3 on the anchor image itself) and applies the *same* absolute margin filter ($s_w - s_l \geq m = 0.05$ after orienting each pair by its score), yielding per-variant pair manifests. Per-variant pre-split pair counts at margin 0.05: GAL-VideoReward $52{,}084$; GAL-HPSv3 $55{,}263$; GAL-Unipercept+HPSv3 $51{,}780$; GAL-Image $58{,}680$; GAL-Unipercept $55{,}797$. One minor confound remains, the training cluster: the three Cluster-A variants (GAL-VideoReward, GAL-HPSv3, GAL-Unipercept+HPSv3, plus the GAL-VideoReward w/o scaffold ablation) ran on $8\times$H100-80G with per-device batch 1 and gradient accumulation 8 (effective global batch 64); the two Cluster-B variants (GAL-Image, GAL-Unipercept) ran on $7\times$RTX-4090 with per-device batch 1 and gradient accumulation 9 (effective global batch 63). Under FSDP2 + bfloat16 we expect the numerics to be approximately matched; the effective-batch delta of 1 between clusters is a real but minor confound. The GAL-VideoReward w/o scaffold ablation shares the *full* GAL-VideoReward configuration (same VideoReward-Overall reward, same $46{,}923$/$5{,}161$ train/val pairs at margin 0.05, same cluster, same effective batch, same hyperparameters) and differs in exactly one field: presence vs. absence of the preset adapter (the `preset_lora_path`/`preset_lora_scale=1.0` keys, scaffolded into the base via deep-copy before reference cloning); see Table 3 and Section 6.5. The full per-variant manifest (output dirs, slurm job IDs, wandb runs/tags, pair manifest basenames,

checkpoint step lists, preset-adapter sha256, and per-variant trainable / total parameter counts) is released as `results/training_configs.json`.

Differential-LoRA scaffold. We train the task LoRA inside a pre-existing differential-LoRA scaffold for Z-Image-Turbo (Ostris, 2025; kelseye.xh & Duan, 2025). At each transformer layer q, let A_q be the frozen preset weight update (the Ostris training adapter) and $B_q(\psi)$ the full task-LoRA weight update. The preset is removed at inference, giving

$$W_q^{\text{train}} = W_q^0 + B_q(\psi) + A_q, \quad W_q^{\text{infer}} = W_q^0 + B_q(\psi). \tag{11}$$

We use the published preset scale `preset_lora_scale` $= 1.0$.[2]

Compute budget (training cluster offline; raw timing logs not retained). Corpus generation is amortized once across all variants. Candidate generation: $49{,}416$ candidate records sampled under the SDE sampler settings ($12{,}354$ prompts $\times 4$) $\times \sim 1.5$ s per candidate (≈ 21 GPU-hours estimated from these per-unit times). I2V rendering: roughly five minutes per clip for HunyuanVideo-1.5 on one RTX 4090 $\times 49{,}416$ candidates ($\approx 4{,}100$ GPU-hours estimated); VideoReward inference is negligible relative to the I2V cost. DPO training on each Cluster-A variant ran the configured schedule of $40{,}000$ maximum steps with $1{,}000$ warmup steps and cosine annealing on $8\times$H100-80G, and the two Cluster-B variants ran on $7\times$RTX-4090; all evaluations use the step-9000 checkpoint, at which the loss had converged. Per-variant DPO wall-times were not logged.

Pair-margin table. Table 5 reports the DPO pair count on the GAL-VideoReward pair manifest as a function of the absolute reward margin threshold m; every variant trains at $m = 0.05$ on its own manifest (per-variant counts above).

Table 5: DPO pair count as a function of the absolute reward margin threshold m on the GAL-VideoReward pair manifest. All reward variants use the same operating point, $m = 0.05$, on their own reward-specific pair manifests (GAL-VideoReward: 52,084 pairs), each landing near a $\approx 5{\times}10^4$-pair budget. A controlled pair-count sweep that holds data volume fixed across margins is listed as future work.

m	Pairs	Note
0.00	74,124	raw, drop ties only
0.05	**52,084**	**operating point**
0.10	33,781	
0.15	20,421	
0.20	11,672	

D Prompt Sources and VBench Prompt Protocol

Prompt sources. The prompt pools used for training and evaluation are disjoint (Table 6). The DPO training pool contains $12{,}354$ prompts sampled from the Lexica prompt corpus (≈ 80k prompts) after canonicalization, exact-duplicate removal, and stripping of non-semantic metadata such as trailing author attributions ("*by <artist>*") and resolution / style tags; these prompts are used only for offline pair construction, and for each one we sample four sibling anchors under the SDE sampler settings, producing the $49{,}416$ candidate records and the shared $74{,}124$-row raw pair pool of Appendix C. The VBench evaluation pool is the official VBench prompt suite (Huang et al., 2024): every open-source I2V model (HunyuanVideo-1.5, LTX-Video, and CogVideoX-1.5) is evaluated on the *full* official VBench per-dimension prompt set for each scored dimension (93 prompts for the imaging-quality / aesthetic-quality / overall-consistency group and 72 for the dynamic-degree / subject-consistency / motion-smoothness group) rather than on any subsampled subset, extended with the fixed video-instruction template described below. With three I2V seeds per anchor, each variant contributes 279 videos to imaging quality, aesthetic quality, and overall consistency and 216 videos to dynamic degree,

[2]This differential-LoRA scaffold is unrelated to the *DiffLoRA* hypernetwork of Wu et al. (2024), which generates personalized LoRA weights with a diffusion model.

subject consistency, and motion smoothness on every open-source renderer. The Kling evaluation pool contains 50 prompts held out from the same Lexica corpus and never used in DPO training; it is restricted to 50 prompts because Kling v3 Pro is a paid closed-source I2V model.

Disjointness audit. We verified pairwise prompt disjointness among the three pools after canonicalization, which applies Unicode normalization, lowercasing, whitespace normalization, punctuation normalization, and removal of the fixed instruction-template wrappers before exact-string comparison. The training and Kling pools are drawn from the same Lexica corpus but form a disjoint train / held-out split, while the VBench pool comes from a different source; the resulting overlap counts are all zero, $|P_{\text{train}} \cap P_{\text{VBench}}| = |P_{\text{train}} \cap P_{\text{Kling}}| = |P_{\text{VBench}} \cap P_{\text{Kling}}| = 0$. Thus the VBench evaluation shares no prompts with the DPO training corpus, and the Kling batch is disjoint from both the training pool and the VBench pool.

Table 6: Prompt sources and disjointness audit. All overlap counts are computed after prompt canonicalization. The training and Kling pools are a disjoint train / held-out split of the Lexica corpus; the VBench pool is the official suite.

Split	# prompts	Source	Overlap with other splits
DPO train	12,354	Lexica corpus (cleaned, deduplicated)	0
VBench eval	93 / 72 per dim	Official VBench per-dimension sets	0
Kling eval	50	Held-out Lexica (disjoint from train)	0

VBench instruction template. Every VBench evaluation in this paper (the HunyuanVideo-1.5 main table and the LTX-Video / CogVideoX-1.5 transfer tables) is driven by the same prompt set: the official VBench prompt suite (Huang et al., 2024) threaded through our pipeline as the conditioning text for the T2I generator, with each prompt extended into a complex-motion video instruction template before scoring. The extension is required because the bare VBench captions are short noun-phrase-style descriptions that under-exercise the dynamic-degree axis on an image-conditioned I2V model; the extension converts each caption into a video instruction of the form "*render* `<caption>` *as a V-second video with [motion-style cue]*," where the motion-style cue is held fixed per VBench category (e.g., "natural camera movement and physically plausible object motion" for the animal / scene categories) and is identical across all variants and all three open-source I2V models, so only the I2V model rendering the anchor varies.

E Anchor-Seed Sensitivity of the Dynamic-Degree Readout

We directly test whether changing the seed-conditioned upstream anchor can flip the downstream dynamic-degree decision. For each of the 72 official VBench dynamic-degree prompts, we sample 12 SDE sibling anchors from the same T2I generator. The downstream I2V seed and sampling configuration are held fixed within each I2V model, so the 12 repeats differ only in the upstream SDE seed that produces the anchor. Each anchor is then rendered with each frozen I2V model and scored by VBench. For I2V model b, prompt p, and anchor seed s, let $d_{b,p,s} \in \{0, 1\}$ be the binary VBench dynamic-degree label. The dynamic fraction $\mathrm{FracDyn}_b$ is the mean of $d_{b,p,s}$ over the 72 prompts and 12 seeds, and the anchor-seed sensitivity Sens_b is the within-prompt variance of the binary label over the 12 anchor seeds, averaged over the 72 prompts (Table 7 reports the population and sample versions). A value of zero means that all 12 seeds agree for every prompt; the variance is largest when the seeds split between static and dynamic outputs. This is a within-prompt statistic, not a pooled variance over all 72×12 clips, since pooling would confound anchor sensitivity with prompt difficulty.

In Table 7, LTX-Video and CogVideoX-1.5 exhibit larger within-prompt variance, meaning that the anchor seed can move these I2V models between static and dynamic outcomes. WAN (Wan Team et al., 2025) shows the opposite pattern: it has the highest dynamic fraction but the smallest

Table 7: Anchor-seed sensitivity of the binary VBench dynamic-degree readout. For each of the 72 official VBench dynamic-degree prompts, we sample 12 seed-conditioned sibling anchors from the same T2I generator and render them with each frozen I2V model under a fixed downstream setting. VBench returns a binary label $d \in \{0, 1\}$ for each rendered clip. We report the overall dynamic fraction and the prompt-averaged within-prompt variance across the 12 seeds. Low variance means that changing the seed-conditioned anchor rarely changes the binary dynamic/static decision for the same prompt. WAN has the highest dynamic fraction and the lowest within-prompt variance, indicating a near-saturated and low-sensitivity dynamic-degree readout.

I2V model	Dynamic fraction ↑	Within-prompt var. (ddof=0) ↓	Within-prompt var. (ddof=1) ↓
LTX-Video	0.5556	0.10204	0.11132
CogVideoX-1.5	0.6481	0.07735	0.08439
HunyuanVideo-1.5	0.7963	0.03318	0.03620
WAN	0.8322	0.02865	0.03125

within-prompt variance. Thus WAN is not a low-motion I2V model; rather, it is already biased toward dynamic outputs and its binary dynamic-degree readout changes little when the anchor changes.

This diagnostic complements the objective-level discussion in Section 5 by testing sensitivity to anchor changes empirically. Low within-prompt variance means that the sampled anchors rarely change the binary dynamic-degree decision. It does not establish weak sensitivity of the rendered video itself: continuous motion changes may remain on the same side of the metric threshold. This finite-sample probe neither estimates a renderer derivative nor covers the other VBench dimensions and reward variants.

F Failure-Mode Case Studies

We provide qualitative case studies of the motion-smoothness regressions in Table 1. All deltas below are GAL-VideoReward relative to Base on the same I2V model; video pairs are released in the supplementary material.

Pattern 1 (texture sharpening; motion smoothness -0.13% on HunyuanVideo-1.5, -0.19% on LTX-Video, -0.86% on CogVideoX-1.5, and -0.41% on Kling v3 Pro). On the inspected prompts with high-frequency texture (e.g., grass, fabric weaves), GAL-VideoReward produces sharper textures than Base, and the corresponding renders show slightly inconsistent edges across frames. This observation suggests that some texture changes may contribute to temporal inconsistency; the objective-level discussion in Section 5 does not establish this mechanism.

Pattern 2 (global camera motion). On the inspected prompts with global camera motion, sharper compositions and higher inter-frame contrast co-occur with less smooth interpolation. These selected cases describe a failure pattern without isolating composition as its cause.

A plausible, untested explanation is that some high-frequency anchor details are rendered inconsistently across frames. This does not make sharpening universally harmful: GAL-VideoReward improves imaging quality and dynamic degree in the aggregate while exhibiting small motion-smoothness regressions. These are distinct evaluation axes, and the case studies do not isolate a causal sharpening effect. A natural mitigation is a reward signal that explicitly penalizes flicker (a temporal-consistency-aware reward), or a regularizer on the high-frequency content of the anchors; both are listed as future work.

G Per-Interval Motion Visualization

The per-interval values in Figure 1 are computed from every decoded frame of each video, without temporal subsampling. Each frame is converted to grayscale and, if its long side exceeds $L = 480$ pixels, downsampled while preserving its aspect ratio. Specifically, for an original height H and

width W, we set

$$s = \min\left(1, \frac{L}{\max(H, W)}\right), \qquad H' = \mathrm{int}(sH), \quad W' = \mathrm{int}(sW), \tag{12}$$

where `int` denotes truncation to an integer; resizing uses the default linear interpolation of OpenCV. Thus the procedure never upsamples an input frame.

Let $\mathbf{F}_n(\mathbf{u}) = (F_n^x(\mathbf{u}), F_n^y(\mathbf{u}))$ denote the dense Farnebäck flow (Farnebäck, 2003) at resized-frame pixel $\mathbf{u} \in \Omega$ from frame $n-1$ to frame n. We use `pyr_scale=0.5`, `levels=3`, `winsize=21`, `iterations=3`, `poly_n=5`, `poly_sigma=1.2`, and `flags=0`. For each adjacent-frame pair, we extract the component-wise spatial median and the mean residual-flow magnitude,

$$\mathbf{g}_n = \left(\underset{\mathbf{u}\in\Omega}{\mathrm{median}}\, F_n^x(\mathbf{u}), \underset{\mathbf{u}\in\Omega}{\mathrm{median}}\, F_n^y(\mathbf{u})\right), \tag{13}$$

$$\mu_n = \frac{1}{|\Omega|} \sum_{\mathbf{u}\in\Omega} \left\|\mathbf{F}_n(\mathbf{u}) - \mathbf{g}_n\right\|_2. \tag{14}$$

The residual magnitude μ_n measures in-scene motion after the dominant translational flow $\mathbf{g}_n$ is removed. For a video with N decoded frames, the $N-1$ adjacent-frame pairs are split into $B = 5$ contiguous intervals of equal length with boundaries $e_b = \mathrm{round}\big((N-1)\, b/B\big)$ for $b = 0, \ldots, B$, and the displayed value of interval b is the mean residual magnitude over its pairs,

$$\bar{\mu}_b = \frac{1}{e_b - e_{b-1}} \sum_{n=e_{b-1}+1}^{e_b} \mu_n, \qquad b = 1, \ldots, B. \tag{15}$$

The inputs used in Figure 1 contain 121 frames at 24 fps for HunyuanVideo-1.5, Kling v3 Pro, and LTX-Video, and 81 frames at 16 fps for CogVideoX-1.5, so every interval spans exactly one second (24 or 16 adjacent-frame pairs) and every adjacent-frame pair contributes to exactly one interval. All values are measured in resized-pixel units per adjacent-frame pair; the script performs neither temporal resampling nor normalization by frame rate or duration, and extraction contains no random sampling. Lines connect the interval means as a visual guide only. Within each I2V-model panel, the three variants share one vertical axis, but different I2V-model panels are scaled independently. Consequently, the visualization supports comparisons among the three variants within an I2V model, not numerical comparisons of motion magnitude across I2V models.

This construction is a diagnostic motion proxy, not a camera-pose estimate or a 3D reconstruction. The median flow $\mathbf{g}_n$ is only a proxy for dominant 2D translational flow and need not isolate camera motion. Likewise, the residual magnitude can respond to object motion, rotation, scaling, parallax, occlusion, flicker, compression artifacts, and optical-flow estimation error. We therefore use these per-interval values only to diagnose the selected examples qualitatively; they do not replace the aggregate quantitative metrics reported in the main evaluation.

H Reward-Model Scores on Transfer I2V Models

Table 8 reports all reward variants on LTX-Video, CogVideoX-1.5, and Kling v3 Pro. These I2V models are excluded from preference construction. The corresponding HunyuanVideo-1.5 results are in Table 4, and Table 2 compares Base with GAL across all four I2V models.

Table 8: Reward-model scores on the three transfer I2V models across GAL variants. VideoReward-Overall scores rendered clips; HPSv3 and UniPercept score their extracted frames. Parentheses give the absolute change from Base; bold marks the best value within each model.

Variant	VideoReward Overall	HPSv3 (video frames)	UniPercept (video frames)
LTX-Video (transfer)			
Base	-0.328	6.4788	50.0465
GAL-Image	-0.763 (-0.435)	7.7409 ($+1.2621$)	55.7652 ($+5.7187$)
GAL-HPSv3	-0.672 (-0.344)	**7.7935** ($+1.3147$)	55.0136 ($+4.9671$)
GAL-Unipercept	-0.448 (-0.120)	7.6155 ($+1.1367$)	**56.3649** ($+6.3184$)
GAL-VideoReward	$\mathbf{-0.224}$ ($+0.103$)	7.3407 ($+0.8619$)	54.1946 ($+4.1481$)
CogVideoX-1.5 (transfer)			
Base	-0.284	5.1392	46.5880
GAL-Image	-0.727 (-0.443)	6.2742 ($+1.1350$)	51.7527 ($+5.1647$)
GAL-HPSv3	-0.716 (-0.432)	6.2377 ($+1.0985$)	50.3643 ($+3.7763$)
GAL-Unipercept	-0.380 (-0.096)	**6.4148** ($+1.2756$)	**53.3527** ($+6.7647$)
GAL-VideoReward	$\mathbf{-0.129}$ ($+0.155$)	6.1601 ($+1.0209$)	51.4183 ($+4.8303$)
Kling v3 Pro (closed-source transfer)			
Base	$+0.394$	8.2367	59.5410
GAL-Image	-0.862 (-1.256)	9.5553 ($+1.3185$)	63.8397 ($+4.2987$)
GAL-HPSv3	-0.160 (-0.554)	**10.5556** ($+2.3189$)	62.4798 ($+2.9389$)
GAL-Unipercept	-0.025 (-0.419)	9.4017 ($+1.1650$)	**69.0857** ($+9.5448$)
GAL-VideoReward	$\mathbf{+0.909}$ ($+0.515$)	8.9403 ($+0.7035$)	62.4870 ($+2.9460$)