FROM PREFERENCE TO FLOW: TURNING PAIRWISE PREFERENCES INTO LOCAL FLOW SUPERVISION

Anonymous authors
Paper under double-blind review

ABSTRACT

Preference optimization has emerged as an effective post-training paradigm for aligning visual generative models with human preferences. Prior methods usually use pairwise preferences as ranking signals over final outputs or as objectives over the generation trajectory, but they rarely turn such preferences into explicit local supervision for intermediate flow dynamics. Motivated by this gap, we propose **P2Flow (Preference-to-Flow)**, a preference post-training framework that converts pairwise preferences into local state–velocity supervision for intermediate flow dynamics. Given a pretrained flow-based generator, P2Flow uses the latent displacement from the preferred output to the rejected one as a pair-specific preference-violating direction to construct perturbed intermediate states around the preferred flow states, and pairs them with corrective velocity targets whose implied clean endpoint remains anchored to the preferred latent. A timestep-dependent modulation further creates bridge-shaped state perturbations that vanish at both the clean and Gaussian endpoints. This approach retains the standard squared velocity-regression objective of flow matching and requires no additional network modules, reward models, or online rollouts during optimization. In extensive experiments using our 40.8K high-confidence human–VLM consensus preference pairs, P2Flow consistently outperforms supervised fine-tuning and strong preference-optimization baselines on Wan2.1-T2V-14B and LingBot-Video-Dense across three benchmarks, showing clear improvements in both automatic metrics and human evaluations.

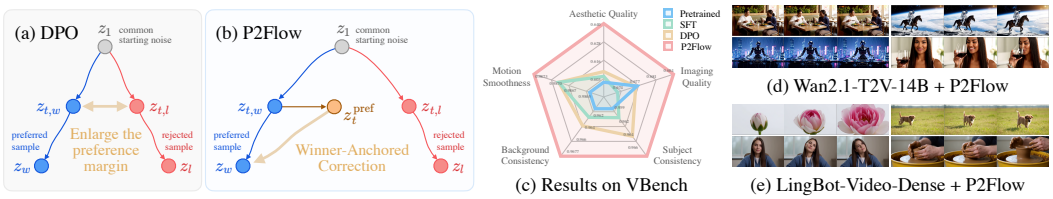
1 INTRODUCTION

Video generative models have advanced rapidly in recent years (Wan, 2025; Seedance, 2026; MiniMax, 2026), with post-training becoming increasingly important for improving generation quality and aligning model outputs with human preferences. Among post-training approaches, Direct Preference Optimization (DPO) (Rafailov et al., 2023; Wallace et al., 2024) provides a simple and scalable paradigm for learning directly from pairwise preference data.

Recent works have extended preference optimization to video generation by designing video-specific objectives (Liu et al., 2025b;a; Li et al., 2026b) and introducing finer temporal or spatio-temporal supervision (Wu et al., 2025; Liu et al., 2026; Huang et al., 2026). Despite these advances, existing methods primarily formulate preference learning through relative objectives over generated outputs or trajectories, while flow matching directly learns velocity targets at sampled intermediate states. The mismatch between preference representation and flow supervision motivates a natural question: *can pairwise preferences be transformed into local state–velocity supervision for flow dynamics?*

Given a preferred and a rejected video under the same text condition, their latent displacement provides more than a relative ranking: it defines a pair-specific direction from the preferred sample toward a less-preferred outcome, which we interpret as a local preference-violating direction. This perspective suggests using pairwise preferences not only to compare generated outputs, but also to construct local state–velocity supervision around the preferred flow trajectory. This formulation offers two advantages. First, it converts pairwise preferences into explicit state–velocity regression targets, providing a more direct and actionable learning signal for the vector field than the indirect margin-enlarging objective used in standard DPO (Xie et al., 2024). Second, it teaches the vec-

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.



(c) Results on VBench

(d) Wan2.1-T2V-14B + P2Flow

(e) LingBot-Video-Dense + P2Flow

Figure 1: Overview of P2Flow. (a–b) Comparison of different preference-learning paradigms. DPO enlarges preference margins, whereas P2Flow learns corrective directions toward preferred outputs. (c) VBench results on Wan2.1-T2V-14B, where P2Flow substantially outperforms prior methods. (d–e) Video examples generated by P2Flow across different backbones.

tor field a corrective behavior for intermediate states perturbed along the winner-to-loser direction, encouraging generation toward higher-quality, preferred outcomes.

Based on this perspective, we propose **P2Flow (Preference-to-Flow)**, a preference post-training framework that converts pairwise preferences into winner-anchored local state–velocity supervision. As illustrated in Figure 1(b), starting from an intermediate state $z_{t,w}$ on the preferred flow trajectory, P2Flow uses the winner-to-loser displacement to define a local preference correction in velocity space and constructs the corresponding perturbed state z_t^{pref} while keeping the preferred sample z_w as the implied clean endpoint. A shared-noise coupling yields a timestep-dependent preference displacement that vanishes toward the Gaussian endpoint. Combined with the winner-anchored state construction, this produces a bridge-shaped state perturbation that vanishes at both flow endpoints. In this way, P2Flow provides a direct and flow-native preference supervision signal while retaining the standard velocity-regression formulation of flow matching. Further analysis shows that the learned vector field exhibits corrective behavior toward higher-quality, preferred outputs.

Existing video preference datasets (Liu et al., 2025b; Li et al., 2026b) largely rely on machine-generated preference labels and videos generated by earlier video generation models. To address these limitations, we introduce a new preference dataset built with systematic category and semantic balancing, state-of-the-art video generators, and strict human–VLM consensus annotation, yielding 40.8K high-confidence preference pairs. Compared with existing datasets, ours offers greater scale, more systematic coverage, greater diversity, and higher annotation quality.

We evaluate P2Flow on two flow-based text-to-video backbones, Wan2.1-T2V-14B (Wan, 2025) and LingBot-Video-Dense (Ma et al., 2026), across VBench (Huang et al., 2024), VideoJAM (Chefer et al., 2025), and our held-out DiverseT2V-Bench. P2Flow consistently improves both backbones and outperforms supervised fine-tuning and existing preference-optimization approaches across visual quality, motion quality, temporal consistency, and text alignment, demonstrating its strong effectiveness. Our contributions are summarized as follows:

- We introduce **P2Flow**, a new preference optimization paradigm that converts winner–loser pairs into winner-anchored local state–velocity supervision, enabling direct and flow-native preference alignment without extra modules, reward models, or online rollouts.
- We build a high-confidence video preference dataset using state-of-the-art models and strict human–VLM annotation, surpassing existing datasets in scale, coverage, diversity, and quality.
- P2Flow consistently improves multiple backbones and outperforms prior preference-optimization methods across diverse benchmarks and human evaluations.

Dataset, model, and code will be released to support further research on preference optimization.

2 Related Work

Text-to-Video Generation. Text-to-video (T2V) generation has advanced rapidly with the development of diffusion-based generative models (Ho et al., 2022; Singer et al., 2022; Blattmann et al., 2023; Chen et al., 2024). Following the success of diffusion transformers (Peebles & Xie, 2023), large-scale T2V models have increasingly adopted transformer backbones (Yang et al., 2025; Ma et al., 2024; Kong et al., 2024; Polyak et al., 2024; Wan, 2025; Ma et al., 2026). Meanwhile, flow matching and rectified flow have emerged as scalable alternatives to conventional diffusion objec-

tives (Lipman et al., 2022; Liu et al., 2022; Esser et al., 2024). These formulations have since been adopted by recent video generators (Jin et al., 2025; Kong et al., 2024; Wan, 2025; Ma et al., 2026).

Preference Alignment for Visual Generative Models. Preference alignment for visual generation has been studied through reward-based learning and direct preference optimization (Kirstain et al., 2023; Xu et al., 2023; Fan et al., 2023; Black et al., 2024; Wallace et al., 2024; Yang et al., 2024; Li et al., 2024). Recent works further extend preference optimization to flow-based generative models (Kim et al., 2024; Wang et al., 2026; Lu et al., 2026; Li et al., 2026a) and video generation with temporal or spatio-temporal supervision (Yuan et al., 2024; Liu et al., 2025b;a; Wu et al., 2025; Huang et al., 2026; Liu et al., 2026; Li et al., 2026b; Cai et al., 2025). In contrast, P2Flow directly converts pairwise latent displacement into winner-anchored local state–velocity supervision for the generative dynamics.

3 Methodology

3.1 Preference-to-Flow (P2Flow)

Preliminaries. Let z_0 denote a clean video latent and $z_1 \sim \mathcal{N}(0, I)$ a Gaussian noise latent. Rectified flow (Liu et al., 2022) defines the linear path $z_t = (1-t)z_0 + tz_1$, $t \in [0, 1]$, with target velocity $v = z_1 - z_0$. The model v_θ is trained to predict v from (z_t, t, c) and generates samples by reversing the learned flow from Gaussian noise to a clean video latent.

Preference Direction. Given a pairwise preference sample (x_w, x_l, c), where x_w is preferred over x_l under the same condition c, we encode the preferred and rejected videos into latent representations as $z_w = E(x_w)$ and $z_l = E(x_l)$, where E denotes the pretrained video encoder. We define the pair-specific winner-to-loser latent displacement as

$$d = z_l - z_w. \tag{1}$$

We use d to characterize the local displacement between the preferred and rejected endpoints in the pretrained latent space. For the preferred latent z_w, the standard rectified-flow state at timestep t is $z_{t,w} = (1-t)z_w + tz_1$, with the corresponding velocity target $v_{\text{std}} = z_1 - z_w$. Using the same Gaussian noise z_1 for the rejected latent z_l, we analogously define $z_{t,l} = (1-t)z_l + tz_1$. Here, shared noise serves only as a training-time coupling and does not require the paired videos to originate from the same generation noise. Using the same Gaussian endpoint for both samples, their latent displacement at timestep t is

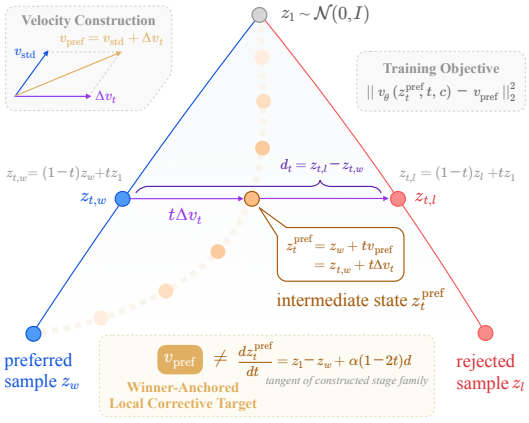


Figure 2: **Illustration of P2Flow.** P2Flow first uses the same-timestep winner-to-loser displacement to construct a preference-guided velocity correction, and then derives the winner-anchored intermediate state while keeping the preferred sample as the implied clean endpoint.

$$d_t = z_{t,l} - z_{t,w} = (1-t)(z_l - z_w) = (1-t)d. \tag{2}$$

Preference-Guided Local Flow Construction. Flow matching directly learns a velocity field. We therefore adopt a velocity-first construction: we first introduce the preference correction into the velocity target and then derive the corresponding intermediate state while keeping the preferred sample as the implied clean endpoint, as illustrated in Figure 2. For each preference pair, we sample a perturbation strength $\alpha \sim \mathcal{U}(0, 1)$ and define the preference correction directly from the same-timestep displacement in Eq. (2):

$$\Delta v_t = \alpha d_t = \alpha(1-t)d. \tag{3}$$

This exposes the model to varying perturbation strengths along the pair-specific winner-to-loser direction. The correction naturally vanishes toward the Gaussian endpoint, where the preferred and

rejected states converge under the shared-noise coupling. The resulting preference-guided velocity target is

$$v_{\text{pref}} = v_{\text{std}} + \Delta v_t = z_1 - z_w + \alpha(1-t)(z_l - z_w). \tag{4}$$

The key insight is to keep the preferred sample z_w as the target clean endpoint. After modifying the velocity to v_{pref}, we therefore construct the intermediate state so that reversing this local flow relation still recovers z_w:

$$z_t^{\text{pref}} = z_w + t v_{\text{pref}}. \tag{5}$$

Substituting Eq. (4) gives

$$\begin{aligned} z_t^{\text{pref}} &= z_w + t\left[z_1 - z_w + \alpha(1-t)(z_l - z_w)\right] \\ &= z_{t,w} + t\alpha(1-t)(z_l - z_w) \\ &= z_{t,w} + t\Delta v_t. \end{aligned} \tag{6}$$

The resulting displacement from the preferred flow state is modulated by $t(1-t)$ in a timestep-dependent manner, forming a bridge-shaped perturbation that vanishes at both flow endpoints. We therefore train on locally rejection-directed perturbations around the preferred flow state with winner-anchored corrective velocity targets.

Preference Flow Objective. Given the constructed preference-violating state z_t^{pref} and its winner-anchored velocity target v_{pref}, we define the preference flow objective as

$$\mathcal{L}_{\text{pref}} = \mathbb{E}\left[\left\| v_\theta\left(z_t^{\text{pref}}, t, c\right) - v_{\text{pref}} \right\|_2^2\right]. \tag{7}$$

This objective keeps the standard squared velocity-regression form of flow matching, while injecting preference information through the constructed state–velocity pair: the rejected sample provides a local perturbation around the preferred flow state, and the preferred sample serves as the implied clean endpoint. Compared with recent video preference methods that require fine-grained temporal annotations (Wu et al., 2025) or online reward-based rollouts (Li et al., 2026c; Kazimi et al., 2026), P2Flow requires no additional annotations, auxiliary reward models, or online sampling, making it simple, lightweight, and readily compatible with existing flow-matching pipelines.

3.2 Analysis of P2Flow

Velocity-First Construction. An alternative is to perturb the intermediate state directly as $\tilde{z}_t = z_{t,w} + \rho(1-t)d$. Under this construction, preserving the preferred latent z_w as the implied clean endpoint requires the corresponding velocity target to satisfy

$$\tilde{v}_t = \frac{\tilde{z}_t - z_w}{t} = v_{\text{std}} + \frac{\rho(1-t)}{t} d. \tag{8}$$

Thus, this state-first construction induces a velocity correction scaled by $(1-t)/t$, which becomes unbounded as $t \to 0$. The proposed z_t^{pref} instead introduces the preference correction directly in velocity space, yielding a bounded correction that is naturally aligned with the velocity-regression formulation of flow matching. We provide further derivation and discussion in Appendix A.3.

Local vs. Global Flow Consistency. P2Flow provides timestep-wise local supervision rather than defining a new global flow trajectory. Because the preference-guided state is constructed independently at each sampled timestep, the collection $\{z_t^{\text{pref}}\}_t$ need not correspond to a rectified-flow trajectory associated with a single modified clean endpoint. Moreover, for fixed z_1, α, and d,

$$\frac{dz_t^{\text{pref}}}{dt} = z_1 - z_w + \alpha(1-2t)d \neq v_{\text{pref}}. \tag{9}$$

Thus, v_{pref} is a winner-anchored local corrective target rather than the tangent velocity of a globally modified trajectory. We further show in Appendix A.5 that the ideal P2Flow target induces a contractive reverse-time correction toward the preferred flow state under a local Euler update.

Table 1: Comparison of publicly released video human-feedback and preference datasets.

Dataset	Size	Short-side Res.	Pairwise	Categorized	Label Source	Source Models
VideoFeedback (2024)	37.6k videos	256–576 px	–	–	Human	Pika, VideoCrafter2, etc.
VideoDPO (2025b)	10k pairs	320 px	✓	–	Automatic	VideoCrafter2
VideoGen-RewardBench (2025a)	25.2k pairs	384–1080 px	✓	✓	Human	Kling 1.5, PixVerse v2, etc.
Ours	**40.8k pairs**	**480–1080 px**	✓	✓	Human + VLM	MiniMax-H3, Seedance 2.0, etc.

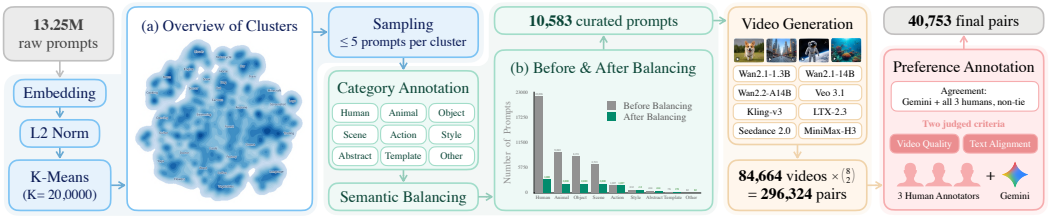


Figure 3: Construction of the proposed video preference dataset.

4 Preference Data and Benchmark Construction

Recent studies show that VLMs enable scalable preference annotation for video generation, but their judgments do not fully coincide with human preferences, particularly in fine-grained evaluation (Wu et al., 2025; Li et al., 2026d; Tong et al., 2025). We therefore combine human judgments with VLM-assisted annotation to construct high-confidence preference data. To further improve the diversity and relevance of the preference data, we generate candidate videos using a broad set of recent text-to-video models, exposing the dataset to varied generation characteristics.

Prompt Collection and Clustering. We collect 13.25M valid prompts from five public video-text sources: InternVid (Wang et al., 2024), Open-Sora-Plan (Lin et al., 2024), OpenVid-1M (Nan et al., 2025), VideoUFO (Wang & Yang, 2026), and Wan2.2-Syn (Zhang et al., 2025a). To organize this large-scale corpus and reduce semantic redundancy, we encode all prompts using Qwen3-Embedding-0.6B (Zhang et al., 2025b) with L2 normalization, cluster the embeddings with K-Means (Douze et al., 2026) using $K = 20{,}000$, and assign each prompt to its nearest cluster by cosine similarity. For visualization, we project the cluster centers to 2D using PCA and t-SNE and visualize their density with WizMap (Wang et al., 2023), with representative keywords illustrating the semantic coverage in Figure 3(a). For each cluster, we use Qwen3.5-35B-A3B (Qwen, 2026) to jointly predict its coarse semantic category, assess its suitability for preference-data construction, and determine its sampling action. Based on these cluster-level annotations, we retain at most five candidate prompts from each retained cluster to reduce redundancy. Details of the cluster-level annotation and candidate construction are provided in Appendix B.4 and Appendix B.5.

Prompt Semantic Balancing. In the preceding semantic categorization step, each retained prompt inherits the coarse semantic category assigned to its cluster: *human*, *animal*, *object*, *scene*, *action*, *style*, *abstract*, *template*, and *other*. As shown in Figure 3(b), the original category distribution is highly imbalanced, with the *human* category accounting for over 40%. Rather than enforcing a uniform distribution, we downsample over-represented categories while preserving broad semantic coverage. We further apply sentence-level quality filtering, as detailed in Appendix B.6. After category-aware balancing and final quality filtering, we obtain 10,583 training prompts.

Video Generation. For each prompt, we generate one candidate video with each of 8 text-to-video models: Wan2.1-T2V-1.3B (Wan, 2025), Wan2.1-T2V-14B (Wan, 2025), Wan2.2-T2V-A14B (Wan, 2025), Veo 3.1 (DeepMind, 2025), Kling v3 (Kling, 2025), LTX-2.3 (HaCohen et al., 2025), Seedance 2.0 (Seedance, 2026), and MiniMax-H3 (MiniMax, 2026). Using generators from different model families and scales exposes the preference dataset to diverse generation characteristics and failure modes. This produces **84,664 candidate videos**.

For each prompt, we exhaustively construct pairwise comparisons between videos generated by different models, yielding $\binom{8}{2} = 28$ comparisons per prompt and **296,324 candidate pairs** in total. The two videos in each pair are conditioned on exactly the same text prompt.

Preference Annotation and Filtering. Each candidate pair is independently evaluated by Gemini-3 and 3 professional human annotators under the same annotation protocol. The presentation order of the two videos is randomized, and the identities of their source generators are hidden. Evaluators compare the two videos along two primary dimensions: *video quality* and *text alignment*. A video is

Table 2: Quantitative comparison across VBench, VideoJAM, and DiverseT2V-Bench. The best results within each backbone and benchmark are highlighted in **bold**.

Method	VBench Metrics					VideoAlign			
	Aesthetic Quality	Imaging Quality	Subject Consistency	Background Consistency	Motion Smoothness	VQ	MQ	TA	Overall
VBench									
Wan2.1-T2V-14B:									
Pretrained	0.6010	0.6769	0.9590	0.9616	0.9864	3.3997	1.6057	3.4826	8.4880
SFT	0.6049	0.6729	0.9603	0.9627	0.9869	3.4385	1.6097	3.5463	8.5945
Vanilla DPO	0.6081	0.6766	0.9630	0.9638	0.9869	3.5877	1.6144	3.6291	8.8312
DenseDPO	0.6091	0.6824	0.9586	0.9635	0.9843	3.4896	1.5790	3.6369	8.7055
ViPO	0.6105	0.6807	0.9620	0.9636	0.9871	3.5660	1.6205	3.7129	8.8994
P2Flow (Ours)	**0.6401**	**0.6844**	**0.9659**	**0.9677**	**0.9873**	**3.6758**	**1.6242**	**4.2898**	**9.5898**
LingBot-Video-Dense:									
Pretrained	0.5857	0.6577	0.9428	0.9471	0.9881	3.5732	1.3206	4.2084	9.1022
SFT	0.5860	0.6589	0.9438	0.9520	0.9876	3.6796	1.3010	4.2448	9.2254
Vanilla DPO	0.5859	0.6566	0.9483	0.9533	**0.9883**	3.7596	1.3213	4.2407	9.3216
DenseDPO	0.5864	0.6576	0.9472	0.9535	0.9882	3.8599	1.3108	4.2708	9.4415
ViPO	0.5851	0.6565	0.9485	0.9511	0.9880	3.7028	1.3237	4.3006	9.3271
P2Flow (Ours)	**0.5999**	**0.6668**	**0.9490**	**0.9539**	**0.9883**	**3.9872**	**1.3328**	**4.4467**	**9.7667**
VideoJAM									
Wan2.1-T2V-14B:									
Pretrained	0.5652	0.6401	0.9321	0.9427	0.9824	2.5413	0.5797	5.7888	8.9098
SFT	0.5771	0.6479	0.9435	0.9496	0.9815	2.6021	0.5833	5.9324	9.1178
Vanilla DPO	0.5768	0.6471	0.9395	0.9480	0.9822	2.6363	0.5842	5.8388	9.0593
DenseDPO	0.5770	0.6480	0.9379	0.9481	0.9795	2.6256	0.5846	5.8781	9.0883
ViPO	0.5784	0.6463	0.9407	0.9500	0.9828	2.5977	0.5958	6.0024	9.1959
P2Flow (Ours)	**0.5965**	**0.6591**	**0.9455**	**0.9509**	**0.9829**	**2.6729**	**0.5971**	**6.3180**	**9.5880**
LingBot-Video-Dense:									
Pretrained	0.5660	0.6599	0.9457	0.9430	0.9882	2.7655	0.6851	6.1412	9.5918
SFT	0.5697	0.6611	0.9475	0.9438	0.9900	2.9072	0.6948	6.1087	9.7107
Vanilla DPO	0.5671	0.6630	0.9490	0.9446	0.9910	2.9207	0.6964	6.1583	9.7754
DenseDPO	0.5659	0.6608	0.9495	0.9455	0.9903	2.9708	0.6921	6.1678	9.8307
ViPO	0.5709	0.6613	0.9496	0.9454	0.9905	2.9344	0.6966	6.1689	9.7999
P2Flow (Ours)	**0.5737**	**0.6662**	**0.9500**	**0.9480**	**0.9913**	**3.1024**	**0.6991**	**6.1755**	**9.9770**
DiverseT2V-Bench									
Wan2.1-T2V-14B:									
Pretrained	0.5673	0.6862	0.9569	0.9547	0.9899	2.8925	0.8757	4.7808	8.5490
SFT	0.5843	0.6958	0.9626	0.9611	0.9903	3.0753	0.9422	5.0682	9.0857
Vanilla DPO	0.5911	0.6940	0.9621	0.9595	0.9905	3.1367	0.9672	5.0266	9.1305
DenseDPO	0.5820	0.6901	0.9597	0.9555	0.9889	3.1410	0.9812	5.1227	9.2449
ViPO	0.5823	0.6847	0.9641	0.9608	0.9906	3.1384	1.0001	5.0178	9.1563
P2Flow (Ours)	**0.6170**	**0.7121**	**0.9652**	**0.9640**	**0.9909**	**3.1463**	**1.0029**	**5.3661**	**9.5153**
LingBot-Video-Dense:									
Pretrained	0.5511	0.6818	0.9522	0.9435	0.9888	3.2711	1.0645	4.8728	9.2084
SFT	0.5597	0.6917	0.9563	0.9504	0.9890	3.2914	1.0740	4.8890	9.2544
Vanilla DPO	0.5489	0.6750	0.9574	0.9491	0.9897	3.3757	1.0568	4.7848	9.2173
DenseDPO	0.5499	0.6752	0.9576	0.9483	0.9899	3.3860	1.0620	4.8773	9.3253
ViPO	0.5537	0.6746	0.9573	0.9490	0.9892	3.2724	1.0316	4.9078	9.2118
P2Flow (Ours)	**0.5658**	**0.6959**	**0.9602**	**0.9523**	**0.9901**	**3.6698**	**1.1203**	**5.1120**	**9.9021**

selected as the winner only when it is judged superior in both dimensions; otherwise, the comparison is treated as a tie. To obtain high-confidence preference supervision, we retain a pair only when Gemini-3 and all 3 human annotators independently select the same non-tie winner. Pairs involving ties or evaluator disagreement are discarded. This strict consensus filtering yields **40,753 high-confidence preference pairs**, corresponding to a 13.75% retention rate. As shown in Table 1, our dataset provides a larger scale, category-aware construction, broader coverage of recent video generators, and high-confidence annotations obtained through strict human–VLM consensus.

DiverseT2V-Bench. We reserve a subset of semantic clusters for evaluation to ensure semantic-level separation from preference training. From these held-out clusters, we sample prompts distinct from the training data and apply the same quality filtering, yielding 178 evaluation prompts.

5 Experiments

5.1 Experimental Setup and Evaluation Protocol

Implementation Details. We evaluate P2Flow on two text-to-video backbones: Wan2.1-T2V-14B (Wan, 2025) and LingBot-Video-Dense (Ma et al., 2026). Both models are fine-tuned with

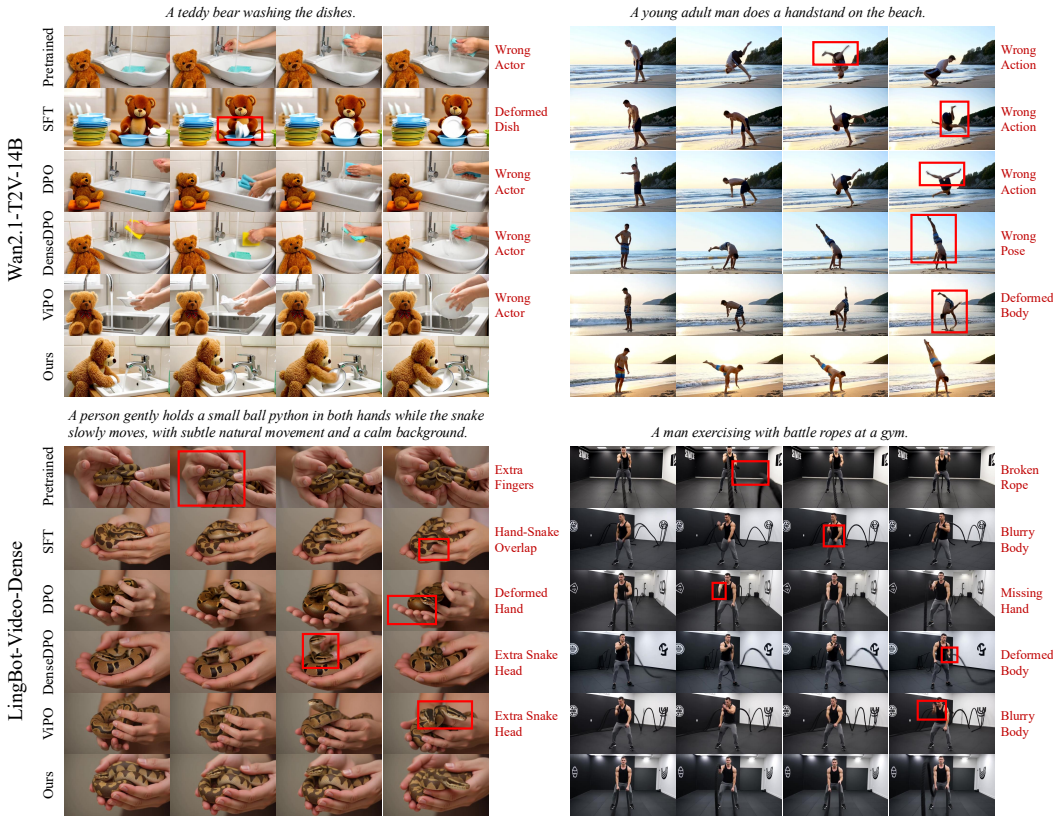


Figure 4: Qualitative comparison results. Across both backbones, P2Flow produces higher-quality videos than other state-of-the-art preference-optimization methods.

rank-32 LoRA using AdamW, with learning rate 5×10^{-5}, weight decay 0.01, and a ConstantLR scheduler. Training uses bfloat16 precision on 8 NVIDIA H20 GPUs.

Compared Methods. We compare P2Flow with the pretrained backbone and four post-training methods: SFT, Vanilla DPO (Liu et al., 2025a), DenseDPO (Wu et al., 2025), and ViPO (Li et al., 2026b). For SFT, we construct a preferred-only subset from the same preference data by selecting one video per prompt that is preferred in at least one retained comparison and is never rejected in any retained comparison. Vanilla DPO, DenseDPO, ViPO, and P2Flow are trained on the same retained preference pairs.

Evaluation Protocol. We evaluate all methods on three prompt sets: (i) 165 prompts from VBench (Huang et al., 2024), following the selection protocol of Huang et al. (2026); (ii) VideoJAM (Chefer et al., 2025); and (iii) our DiverseT2V-Bench. We report five VBench metrics covering complementary aspects of video quality: *Aesthetic Quality*, *Imaging Quality*, *Subject Consistency*, *Background Consistency*, and *Motion Smoothness*. We additionally use VideoAlign (Liu et al., 2025a) to evaluate *Visual Quality* (VQ), *Motion Quality* (MQ), *Text Alignment* (TA), and *Overall Quality*. Higher values indicate better performance.

5.2 Results and Comparisons

Quantitative Comparison. As shown in Table 2, P2Flow consistently achieves the best overall performance across VBench, VideoJAM, and DiverseT2V-Bench. It outperforms the pretrained model, SFT, and existing preference-optimization methods on both Wan2.1-T2V-14B and LingBot-Video-Dense, demonstrating the effectiveness and generality of our preference-to-flow supervision across different benchmarks and backbone models.

Qualitative Comparison. As shown in Figure 4, P2Flow produces higher-quality videos with more coherent structures, more stable motion, and better correspondence to the input prompts. In

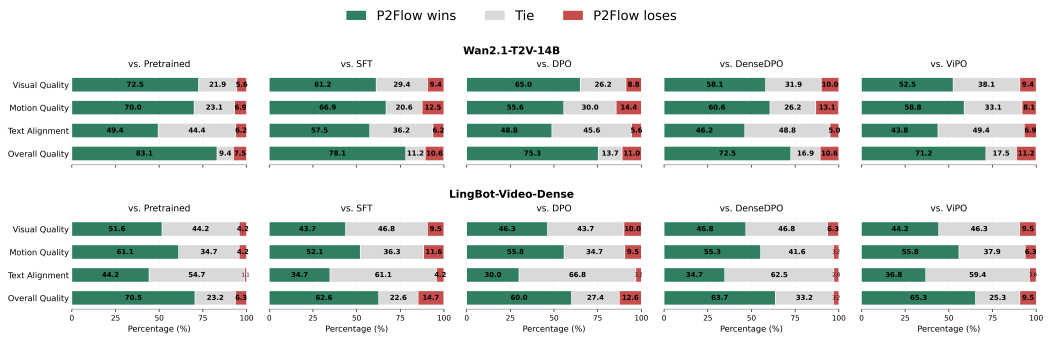


Figure 5: Human evaluation results. We report the percentage of annotations in which P2Flow wins, ties with, or loses to each compared method. P2Flow outperforms all competing methods.

comparison, the pretrained model and existing preference-optimization methods may exhibit structural distortions, appearance inconsistency, or missing prompt details. P2Flow better preserves object appearance and temporal consistency while faithfully realizing the requested visual content, which is consistent with the improvements observed in the quantitative evaluation.

Although this paper focuses on video generation, we show how P2Flow extends to **image generation** in Appendix C. Quantitative and qualitative results demonstrate that P2Flow also outperforms existing methods for image generation, highlighting its effectiveness and robustness.

Human Evaluation. Following prior pairwise user-study protocols (Huang et al., 2026; Wu et al., 2025), we evaluate 50 randomly sampled prompts per backbone. For each prompt, 20 annotators compare anonymized and randomly ordered video pairs in terms of visual quality, motion quality, text alignment, and overall quality, with win/tie/loss judgments. Figure 5 shows that P2Flow consistently achieves higher preference rates than all baselines across the four dimensions.

5.3 Ablation Studies and Analysis

Ablation Studies. We ablate two key designs of P2Flow: the preference direction and timestep modulation. For the preference direction, we compare the designed direction $d_t = z_{t,l} - z_{t,w}$ with four controlled alternatives. *Random* uses a norm-matched Gaussian direction $d_t^{\text{rand}} = \frac{\|d_t\|_2}{\|r\|_2+\epsilon} r$ with $r \sim \mathcal{N}(0, I)$; *Shuffled* uses a winner-to-loser direction from another pair, $\bar{d}_t$, and sets $d_t^{\text{shuf}} = \frac{\|d_t\|_2}{\|\bar{d}_t\|_2+\epsilon} \bar{d}_t$; *Same-prompt Ambiguous* uses $d_t^{\text{amb}} = \frac{\|d_t\|_2}{\|z_{t,a}-z_{t,w}\|_2+\epsilon}(z_{t,a} - z_{t,w})$, where $z_{t,a}$ is an ambiguous candidate from the same prompt whose comparison with the winner is not retained as a high-confidence non-tie preference; and *Reverse* flips the original direction, $d_t^{\text{rev}} = -d_t$. For timestep modulation, we generalize $\Delta v_t = \alpha g(t) d$ and compare $g(t) \in \{1, \sqrt{1-t}, 1-t, (1-t)^2\}$, with $g(t) = 1 - t$ as the default.

As shown in Table 3, the original winner-to-loser direction outperforms all controlled alternatives, while the default linear timestep modulation achieves the best overall performance among the tested schedules. These results support both the pair-specific preference direction and the timestep scaling induced by shared-noise coupling. Corresponding qualitative comparisons are shown in Figure 6.



Figure 6: Qualitative comparison of ablation settings, using the same symbols as in Table 3.

Winner Endpoint Recovery. We evaluate winner-anchored consistency on 500 preference pairs held out from post-training using Wan2.1-T2V-14B at $t \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$. Given z_t^{pref}, we

Table 3: Ablation studies on preference direction and timestep modulation on DiverseT2V-Bench using Wan2.1-T2V-14B. The best results are highlighted in **bold**.

Variant	VBench					VideoAlign				No.
	Aesthetic Quality	Imaging Quality	Subject Consistency	Background Consistency	Motion Smoothness	VQ	MQ	TA	Overall	
Preference Direction										
Random	0.5910	0.6904	0.9635	0.9612	0.9883	3.0244	0.9637	5.1309	9.1190	①
Shuffled	0.5893	0.6824	0.9592	0.9578	0.9860	2.9291	1.0005	5.0235	8.9531	②
Same-prompt Ambiguous	0.5922	0.6871	0.9638	0.9617	0.9890	3.0812	0.9769	5.2436	9.3017	③
Reverse	0.5906	0.6838	0.9630	0.9581	0.9892	3.0124	0.9152	5.0995	9.0271	④
Timestep Modulation										
$g(t) = 1$	0.4457	0.4367	0.9321	0.9497	0.9872	1.3812	0.4546	1.2821	3.1179	⑤
$g(t) = \sqrt{1-t}$	0.5483	0.6324	0.9563	0.9553	0.9872	2.5421	0.7796	3.8645	7.1862	⑥
$g(t) = (1-t)^2$	0.5866	0.6885	0.9634	0.9574	0.9904	3.0147	0.9052	5.1521	9.0720	⑦
P2Flow (Ours)	**0.6170**	**0.7121**	**0.9652**	**0.9640**	**0.9909**	**3.1463**	**1.0029**	**5.3661**	**9.5153**	⑧

recover the implied clean endpoint as $\hat{z}_{0,\theta} = z_t^{\text{pref}} - t\, v_\theta(z_t^{\text{pref}}, t, c)$ and measure

$$E_{\text{win}}(t) = \frac{\|\hat{z}_{0,\theta} - z_w\|_2}{\|z_l - z_w\|_2 + \epsilon}, \qquad (10)$$

where lower values indicate better recovery of the preferred endpoint relative to the winner–loser distance. Since $z_t^{\text{pref}} - t v_{\text{pref}} = z_w$, this error provides a diagnostic of consistency with the winner-anchored local velocity target, as detailed in Appendix A.5. Figure 7 shows that P2Flow achieves the lowest error across all timesteps, while the other post-training methods remain relatively close, suggesting that generic preference optimization does not induce the specific winner-anchored consistency.

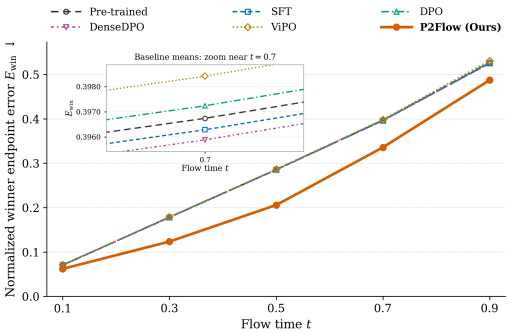


Figure 7: Winner endpoint recovery. Lower E_{win} indicates better agreement with the intended supervision.

Sample Correction. We further evaluate the ability of different models to correct unsatisfactory samples. Specifically, we take rejected samples that are excluded from the final dataset, perturb them with varying levels of noise, and assess how well each model recovers high-quality samples. As shown in Figure 8, P2Flow achieves the best performance across timesteps and consistently improves upon the original inputs. These results indicate that the proposed learning paradigm enables P2Flow to effectively refine unsatisfactory samples, thereby validating the motivation behind our method design.

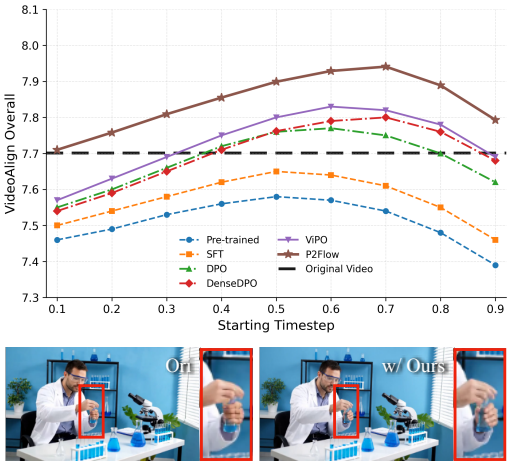


Figure 8: Correction of unsatisfactory samples.

6 Conclusion

We proposed **P2Flow**, a preference post-training method that converts pairwise video preferences into winner-anchored local flow supervision. By jointly constructing preference-violating intermediate states and corresponding velocity targets, P2Flow uses rejected samples to define local deviations while preserving preferred samples as the implied clean endpoints. Experiments across two flow-based video-generation backbones and multiple benchmarks demonstrate consistent improvements over supervised fine-tuning and existing preference-optimization methods. These results suggest that pairwise preferences can provide supervision beyond relative ranking, directly informing the local dynamics of flow-based video generation.

AI USE STATEMENT

We used generative AI tools only to assist with language refinement, including grammatical correction and improvements to the clarity, fluency, and academic presentation of the manuscript. The research methodology, experimental work, and scientific analysis were carried out independently by the authors.

ETHICS STATEMENT

This work involves human annotation of model-generated videos for preference-data construction and evaluation. Human annotators assess video quality and text alignment following predefined annotation protocols. The proposed dataset is constructed from publicly available prompt sources and model-generated videos. Any public release of the dataset will follow the applicable licenses and terms of use of the underlying data sources and models.

REPRODUCIBILITY STATEMENT

We provide the complete formulation of P2Flow and its training objective in the main paper, with additional derivations and theoretical analysis in the appendix. The experimental setup specifies the model backbones, optimization settings, baselines, and evaluation protocols. The appendix further documents the preference-data construction pipeline, including prompt collection and clustering, filtering, candidate video generation, annotation protocols and prompts, and dataset statistics. We will release the dataset, models, and code to facilitate reproduction and future research.

REFERENCES

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

Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *Computer Vision and Pattern Recognition*, 2023.

Yuanhao Cai, Kunpeng Li, Menglin Jia, Jialiang Wang, Junzhe Sun, Feng Liang, Weifeng Chen, Felix Juefei-Xu, Chu Wang, Ali Thabet, et al. Phygdpo: Physics-aware groupwise direct preference optimization for physically consistent text-to-video generation. *ArXiv preprint*, 2025.

Hila Chefer, Uriel Singer, Amit Zohar, Yuval Kirstain, Adam Polyak, Yaniv Taigman, Lior Wolf, and Shelly Sheynin. Videojam: Joint appearance-motion representations for enhanced motion generation in video models. *ArXiv preprint*, 2025.

Haoxin Chen, Yong Zhang, Xiaodong Cun, Menghan Xia, Xintao Wang, Chao Weng, and Ying Shan. Videocrafter2: Overcoming data limitations for high-quality video diffusion models. In *Computer Vision and Pattern Recognition*, 2024.

Team DeepMind. Veo: a text-to-video generation system, 2025. URL https://storage.googleapis.com/deepmind-media/veo/Veo-3-Tech-Report.pdf.

Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. The faiss library. *IEEE Transactions on Big Data*, 12(2):346–361, 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.

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. In *Neural Information Processing Systems*, 2023.

Yoav HaCohen, Benny Brazowski, Nisan Chiprut, Yaki Bitterman, Andrew Kvochko, Avishai Berkowitz, Daniel Shalem, Daphna Lifschitz, Dudu Moshe, Eitan Porat, Eitan Richardson, Guy Shiran, Itay Chachy, Jonathan Chetboun, Michael Finkelson, Michael Kupchick, Nir Zabari, Nitzan Guetta, Noa Kotler, Ofir Bibi, Ori Gordon, Poriya Panet, Roi Benita, Shahar Armon, Victor Kulikov, Yaron Inger, Yonatan Shiftan, Zeev Melumian, and Zeev Farbman. Ltx-2: Efficient joint audio-visual foundation model. *ArXiv preprint*, 2025.

Xuan He, Dongfu Jiang, Ge Zhang, Max Ku, Achint Soni, Sherman Siu, Haonan Chen, Abhranil Chandra, Ziyan Jiang, Aaran Arulraj, et al. Videoscore: Building automatic metrics to simulate fine-grained human feedback for video generation. In *Conference on Empirical Methods in Natural Language Processing*, 2024.

Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. In *Neural Information Processing Systems*, 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 *Computer Vision and Pattern Recognition*, 2024.

Zitong Huang, Kaidong Zhang, Yukang Ding, Chao Gao, Rui Ding, Ying Chen, and Wangmeng Zuo. Mind the generative details: Direct localized detail preference optimization for video diffusion models. In *Computer Vision and Pattern Recognition*, 2026.

Yang Jin, Zhicheng Sun, Ningyuan Li, Kun Xu, Hao Jiang, Nan Zhuang, Quzhe Huang, Yang Song, Yadong Mu, and Zhouchen Lin. Pyramidal flow matching for efficient video generative modeling. In *International Conference on Learning Representations*, 2025.

Tahira Kazimi, Connor Dunlop, and Pinar Yanardag. Diverse video generation with determinantal point process-guided policy optimization. In *Computer Vision and Pattern Recognition*, 2026.

Minu Kim, Yongsik Lee, Sehyeok Kang, Jihwan Oh, Song Chong, and Se-Young Yun. Preference alignment with flow matching. In *Neural Information Processing Systems*, 2024.

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. In *Neural Information Processing Systems*, 2023.

Team Kling. Kling-omni technical report. *ArXiv preprint*, 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*, 2024.

Kesong Li, Yixuan Xu, Kuo-kun Tseng, Weiyi Lu, Kan Liu, and Tao Lan. Linear-dpo: Linear direct preference optimization for diffusion and flow-matching generative models. *ArXiv preprint*, 2026a.

Ming Li, Jie Wu, Jiaxing Cui, Xiaojie Li, Rui Wang, and Chen Chen. Vipo: Visual preference optimization at scale. In *International Conference on Learning Representations*, 2026b.

Rui Li, Yuanzhi Liang, Ziqi Ni, Haibing Huang, Chi Zhang, and Xuelong Li. Rethinking reward signals in video grpo: When scores become targets. In *European Conference on Computer Vision*, 2026c.

Shufan Li, Konstantinos Kallidromitis, Akash Gokul, Yusuke Kato, and Kazuki Kozuka. Aligning diffusion models by optimizing human utility. In *Neural Information Processing Systems*, 2024.

Zhikai Li, Xuewen Liu, Wangbo Zhao, Pan Du, Kaicheng Zhou, Qingyi Gu, Yang You, Zhen Dong, Kurt Keutzer, et al. K-sort eval: efficient preference evaluation for visual generation via corrected vlm-as-a-judge. In *International Conference on Learning Representations*, 2026d.

Bin Lin, Yunyang Ge, Xinhua Cheng, Zongjian Li, Bin Zhu, Shaodong Wang, Xianyi He, Yang Ye, Shenghai Yuan, Liuhan Chen, et al. Open-sora plan: Open-source large video generation model. *ArXiv preprint*, 2024.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *ArXiv preprint*, 2022.

Henglin Liu, Fangyuan Kong, Jing Wang, Yizhou Lin, Nisha Huang, Chang Liu, Xintao Wang, Pengfei Wan, Kun Gai, and Xiu Li. Temporal concentration from rollout errors: Implicit preference optimization for text-to-video diffusion. *ArXiv preprint*, 2026.

Jie Liu, Gongye Liu, Jiajun Liang, Ziyang Yuan, Xiaokun Liu, Mingwu Zheng, Xiele Wu, Qiulin Wang, Menghan Xia, Xintao Wang, et al. Improving video generation with human feedback. In *Neural Information Processing Systems*, 2025a.

Runtao Liu, Haoyu Wu, Ziqiang Zheng, Chen Wei, Yingqing He, Renjie Pi, and Qifeng Chen. Videodpo: Omni-preference alignment for video diffusion generation. In *Computer Vision and Pattern Recognition*, 2025b.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *ArXiv preprint*, 2022.

Yunhong Lu, Qichao Wang, Hengyuan Cao, Xiaoyin Xu, and Min Zhang. Offline preference optimization for rectified flow with noise-tracked pairs. In *International Conference on Machine Learning*, 2026.

Shuailei Ma, Jiaqi Liao, Xinyang Wang, Jingjing Wang, Chaoran Feng, Zijing Hu, Chong Bao, Zichen Xi, Yuqi Gan, Weisen Wang, et al. Scaling mixture-of-experts video pretraining for embodied intelligence. *ArXiv preprint*, 2026.

Xin Ma, Yaohui Wang, Xinyuan Chen, Gengyun Jia, Ziwei Liu, Yuan-Fang Li, Cunjian Chen, and Yu Qiao. Latte: Latent diffusion transformer for video generation. *ArXiv preprint*, 2024.

MiniMax. Minimax-h3, 2026. URL https://github.com/MiniMax-AI/MiniMax-H3.

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. In *International Conference on Learning Representations*, 2025.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *International Conference on Computer Vision*, 2023.

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*, 2024.

Team Qwen. Qwen3.5: Towards native multimodal agents, 2026. URL https://qwen.ai/blog?id=qwen3.5.

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. In *Neural Information Processing Systems*, 2023.

Team Seedance. Seedance 2.0: Advancing video generation for world complexity. *ArXiv preprint*, 2026.

Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, et al. Make-a-video: Text-to-video generation without text-video data. *ArXiv preprint*, 2022.

Haibo Tong, Zhaoyang Wang, Zhaorun Chen, Haonian Ji, Shi Qiu, Siwei Han, Kexin Geng, Zhongkai Xue, Yiyang Zhou, Peng Xia, et al. Mj-video: Benchmarking and rewarding video generation with fine-grained video preference. In *Neural Information Processing Systems*, 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 *Computer Vision and Pattern Recognition*, 2024.

Team Wan. Wan: Open and advanced large-scale video generative models. *ArXiv preprint*, 2025.

Shaomeng Wang, He Wang, Longquan Dai, and Jinhui Tang. Pc-flow: Preference alignment in flow matching via classifier. In *AAAI Conference on Artificial Intelligence*, 2026.

Wenhao Wang and Yi Yang. Videoufo: A million-scale user-focused dataset for text-to-video generation. In *Neural Information Processing Systems*, 2026.

Yi Wang, Yinan He, Yizhuo Li, Kunchang Li, Jiashuo Yu, Xin Ma, Xinhao Li, Guo Chen, Xinyuan Chen, Yaohui Wang, et al. Internvid: A large-scale video-text dataset for multimodal understanding and generation. In *International Conference on Learning Representations*, 2024.

Zijie J Wang, Fred Hohman, and Duen Horng Chau. Wizmap: Scalable interactive visualization for exploring large machine learning embeddings. In *Annual Meeting of the Association for Computational Linguistics*, 2023.

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. In *Neural Information Processing Systems*, 2025.

Shiming Xie, Hong Chen, Fred Yu, Zeye Sun, Xiuyu Wu, and Yingfan Hu. Minor dpo reject penalty to increase training robustness. *ArXiv preprint*, 2024.

Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. Imagereward: Learning and evaluating human preferences for text-to-image generation. In *Neural Information Processing Systems*, 2023.

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 *Computer Vision and Pattern Recognition*, 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*, 2025.

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 *Computer Vision and Pattern Recognition*, 2024.

Peiyuan Zhang, Yongqi Chen, Runlong Su, Hangliang Ding, Ion Stoica, Zhengzhong Liu, and Hao Zhang. Fast video generation with sliding tile attention. *ArXiv preprint*, 2025a.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. Qwen3 embedding: Advancing text embedding and reranking through foundation models. *ArXiv preprint*, 2025b.