Recreating Video Arenas via Automated Preference Scoring

Yue Zhao **Aniket Gupta** **Juze Zhang** **Tiange Xiang**
Ryan Rong **Xuan Su** **Li Fei-Fei** **Ehsan Adeli**

Stanford University

Abstract

Evaluating video generation models is notoriously challenging. Existing metrics either fail to align with human judgment or rely on costly, unscalable human ratings. In this work, we introduce **Video Preference Score** (VPS), a fully automated video preference score designed to directly model human judgment and accurately recreate large-scale, Elo-based video arenas. Trained on a newly collected dataset that augments generated videos with retrieved real-world footage, VPS evaluates paired videos against complex text prompts to predict human preference. Experiments demonstrate that VPS correlates highly with human consensus and generalizes zero-shot across unseen models and benchmarks. Furthermore, we showcase the utility beyond evaluation: as a plug-in reward model, VPS seamlessly integrates with Flow-GRPO to improve WAN-2.1's generative quality by 51 Elo points. Overall, VPS provides a scalable and generalizable solution for both evaluating and aligning modern video generation models.

1 Introduction

Video generation models have achieved tremendous progress and have shown promising capabilities in visual reasoning [25, 32], world simulation [3, 37], and making real-world decisions [13, 23, 38]. However, evaluating video generation models remains difficult despite the availability of various evaluation metrics. Existing evaluation metrics fall mainly into two categories: *rule-based* and *rating-based*. Rule-based metrics [12, 15, 26] compose arithmetic operations from pre-trained visual features. These metrics are clearly defined, deterministic, and reproducible. However, many of them require reference input, which is difficult to obtain for open-world video generation. In addition, rule-based metrics often align poorly with human study, since visual perception is, by its nature, subjective. Rating-based metrics [10, 9, 36] instruct human annotators to assign a rating given certain criteria. However, human annotation is costly and time-consuming. The evaluation result is sensitive to the size and representativeness of the participants. Meanwhile, evaluation details are glossed over in most technical reports. As a result, human evaluations are often considered hard to replicate.

In this paper, instead of introducing a brand-new benchmark, we start from an online video arena leaderboard and ask the following question: *How well can we recreate a video arena with minimal effort?* Given a fixed set of text prompts and videos generated by several video models, we can rebuild an Elo-based leaderboard by rerunning video-to-video comparisons based on any independent or pairwise score. The correlation between the recreated leaderboard and the public leaderboard then tells us how well the used video score aligns with human judgments.

To this end, we propose a fully automatic evaluation system that measures alignment with the human preferences of a large crowd, illustrated in Figure 1. We prioritize human preference and propose to model it directly. The proposed **video preference score**, or VPS for short, takes as input paired videos

Data, code and models are available here.

40th Conference on Neural Information Processing Systems (NeurIPS 2026). Track on Evaluations and Datasets.

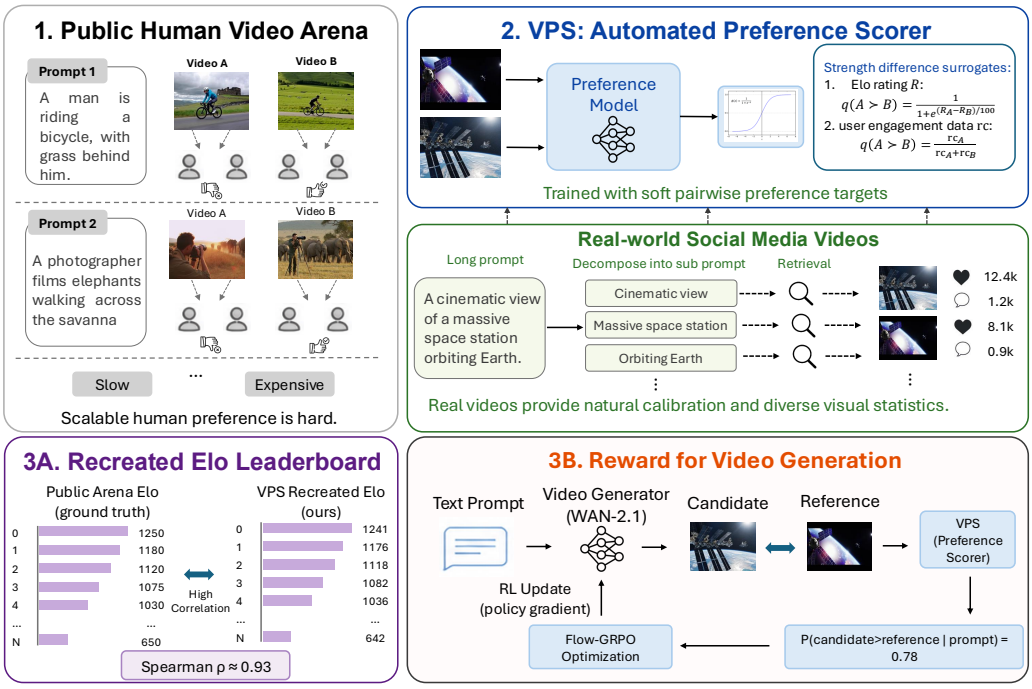


Figure 1: **Overview.** VPS is an automated preference score to approximate public video arenas based on large-scale human votes. VPS can recreate the Elo-based leaderboard with high correlation (Section 5.2) and serve as a reward model for reinforcement fine-tuning a video generation model (Section 5.3).

and a text prompt, and outputs a scalar value representing how likely one video is preferred over the other. To learn VPS, we construct a dataset with both generated videos from the up-to-date video generation models and real-world videos from social media. We propose a data collection pipeline to retrieve highly relevant videos given long, complicated text prompts, which are challenging but more commonly used in modern video evaluation benchmarks.

We summarize our key findings as follows. First, VPS aligns with human preference on the Video Arena leaderboard, outperforming existing rule-based and rating-based metrics. Second, the learned human preference score generalizes across models and benchmarks. We show that the VPS learned from a subset of video generation models correctly predicts the preference order of unseen models with a correlation of 0.710. VPS learned from the Artificial Analysis leaderboard remains robust on Arena AI. Third, real-world videos serve as a good source to augment the video preference data. Relative social-media engagement between two real videos retrieved for the same prompt provides a weak but freely available preference signal, and adding social-video data improves downstream reinforcement fine-tuning even though it slightly weakens arena-only reconstruction. Finally, we show that the preference score serves as a plug-in reward model and improves the quality of video generation. Flow-GRPO with VPS shows a quantitative improvement of 51 Elo points. Qualitative results also support our findings.

2 Related Work

Video Evaluation. Video evaluation is multifaceted, and many evaluation metrics have been proposed. Fréchet Video Distance (FVD) [26] measures the distributional distance between the real-world data and the generative model, both of which are approximated with multivariate Gaussians from some pre-trained feature embeddings, following FID [11] in images. FVD, as a metric, is often criticized by (1) an impractically large sample size required for reliable estimation [20], (2) an underrepresentative pretrained video encoder [4], and (3) bias towards static scenes [7]. ViCLIP [28], T2VScore [34], and ETVA [8] emphasize text-to-video alignment while ignoring other dimensions. Video Quality

Assessment metrics [31, 33] mainly capture the aesthetic aspect of videos. More comprehensive benchmarks, such as VBench [12] and EvalCrafter [19], evaluate video generation quality in multiple dimensions by putting together the aforementioned metrics and others. However, their reliance on external pre-trained models increases evaluation costs and makes end-to-end optimization infeasible. Inspired by the success of open platform [5] to evaluate LLMs, crowd-sourced, voting-based video arenas have been built. The rankings are so far the most reliable and more widely recognized However, thousands of human votes are required for each model, making it hard to replicate. Our work confirms the evaluation consistency of these voting-based leaderboards and also proposes a purely computational approach to recreate the arena without repeated human effort.

Human Preference Models. Human preference is a simple and effective objective to solve complex tasks when the reward function is ill-defined [6, 1, 22]. Popularized in LLMs [42], human preference is later adopted to evaluate and improve image generation models, including PickScore [14], MPS [41], HPS [35, 21]. It is a natural extension to improve video generation with human feedback. VideoScore [10] proposes one of the first large datasets with human ratings for generative videos. VideoScore2 [9] further includes detailed chain-of-thought rationales. However, recent studies [18, 36] reveal that pointwise ratings do not align well with ranking-based human preference. Our VPS directly models pairwise preference, resulting in higher correlation. VideoReward [18] defines fine-grained multi-dimensional human preferences and learns a reward model by fine-tuning a vision language model. VisionReward [36] further increases the evaluation dimensions from 5 to 20. Although fine-grained assessment improves interpretability, the multi-dimensional evaluation unavoidably introduces new complexity in determining the final preference from multiple axes.

3 Data Collection

We construct a pairwise preference dataset to learn automatic preference score. This video preference score can be used for reconstructing the video arena without laborious human votings and serve as a reward model for reinforced fine-tuning video models. In order to adhere to the arena-style ranking, we build the dataset with two main design goals. First, the dataset should be large enough to support arena-style model comparison without collecting a new round of human annotations. Second, the data should contain realistic visual statistics that are missing in most existing benchmarks which focus on generated videos only. To this end, we combine generated videos from a public video arena with real-world social-media videos retrieved for the same prompts. The resulting dataset supports three types of comparisons: generated–generated, real–real, and real–generated.

3.1 Prompts

We collect prompts from two existing sources. The first consists of short prompts similar to VBench [12], which typically describe a single scene or action. The second consists of longer paragraph-style prompts from modern video-generation benchmarks, in particular, the Artificial Analysis video arena[1]. These longer prompts are more representative of current text-to-video use cases: they often contain multiple entities, attributes, scene constraints, temporal relations, and stylistic requirements.

3.2 Generated videos from video arenas

We scrape generated videos from the Artificial Analysis video arena. For each example, we store the prompt, the two generated videos, the corresponding models that produced them, and the public Elo score of the corresponding models. These model-level Elo scores provide weak but scalable supervision: if two videos are generated by models with different Elo scores, the video from the higher-Elo model is treated as more likely to be preferred, with the strength of preference determined by the Elo gap rather than by a hard binary label. This lets us convert an existing public arena into a training set for pairwise preference prediction, the details of which are provided in the following section. The Artificial Analysis subset contains 7,372 generated videos corresponding to 694 different prompts. Videos generated under the same input prompt can be paired arbitrarily, resulting in 27,498 pairwise examples.

[1] `https://artificialanalysis.ai/video/arena`

3.3 Real-world videos from social media

Generation-only datasets have an important limitation: their visual diversity and quality are limited by the set of models available at collection time. In contrast, real-world social-media videos contain natural camera motion, lighting, physical dynamics, compression artifacts, human activity, and long-tail scenes that are difficult for current generators to cover. We therefore augment arena data with real videos retrieved from social media.

We develop a social media retrieval API that takes a query string or a list of keywords, and returns relevant video URLs, metadata (such as the observed engagement with the video, including user likes and comments), and the timestamp of the segment judged most relevant to the query. The timestamp is important because full social-media videos often contain intros, transitions, or off-topic segments.

Retrieval for short prompts. For short VBench-style prompts, direct retrieval works reasonably well. A query such as "a cat jumps onto a couch" closely resembles the search queries used by social platforms, and the top returned videos are usually semantically tight.

Long prompts require decomposition. In contrast, Artificial Analysis-style prompts are longer and harder. A single paragraph-level query often contains several semantic components, and social media search may match only the most salient entity while ignoring the action, temporal constraint, or scene attribute. For example, a prompt describing "Machu Picchu at sunrise with mist rising from the valleys" may retrieve generic Machu Picchu videos while missing the sunrise or mist components.

To address this, we decompose each long prompt $\boldsymbol{y}$ into a set of shorter sub-prompts $\{q_i\}_{i=1}^{m}$. Each sub-prompt targets one major semantic component of the original prompt, such as an object, location, action, visual attribute, or temporal condition. We query the retrieval API independently with each sub-prompt and aggregate the returned videos.

Let $\mathcal{R}_i$ denote the ranked set of videos returned for sub-prompt q_i. We form the candidate pool $\mathcal{U}(\boldsymbol{y}) = \bigcup_{i=1}^{m} \mathcal{R}_i$. Each candidate video $v \in \mathcal{U}(\boldsymbol{y})$ is assigned an aggregation score $s_{\mathrm{ret}}(v; \boldsymbol{y}) = \sum_{i=1}^{m} \mathbf{1}[v \in \mathcal{R}_i] - \lambda \sum_{i:v\in\mathcal{R}_i} \mathrm{rank}_i(v)$, where the first term rewards videos retrieved by multiple sub-prompts and the second term breaks ties using the API-provided ranking. We then retain the top videos under s_{ret}. This favors videos that match multiple components of the prompt rather than videos that only match a single keyword.

Prompt decomposition improves real-world retrieval. Table 1 reports a retrieval ablation. We use the FVD score between the set of retrieved videos and generated videos as a proxy to measure the retrieval quality. For short VBench-style prompts, using relevance ranking produces a lower FVD score than random retrieval. The improvement is more significant for long prompts, where top-ranked videos from chunked retrieval reduce FVD from 100 to 45 relative to random sampling from the retrieved pool. These results support our use of prompt decomposition and relevance-ranked aggregation when constructing real-world anchors for long-form video-generation prompts.

Table 1: Retrieval ablation. Prompt decomposition substantially improves real-world retrieval for long prompts. Lower FVD is better.

Prompt set	decomposition	ranking	$\mathrm{FVD}_{\downarrow}$
VBench			60
VBench		✓	54
AA	✓		100
AA	✓	✓	45

3.4 Data split & statistics

Data splits. We have a total of 694 prompts, and split train/val/test at the prompt level by default: no prompt appears in more than one of train, validation, or test. The *prompt-based data split* prevents leakage through repeated prompts and near-duplicate prompt templates. In addition, we experiment with *model-based data split*, ensuring that videos generated by certain models are never seen during training. This is useful to see whether the learned preference score generalizes to newer unseen models, which are likely to be introduced constantly.

Data statistics. Table 2 shows the statistics of the proposed VPS dataset (live at huggingface). By including both generated and real videos, the entire VPS dataset is an order of magnitude larger than existing preference-based datasets, *e.g.*, VideoGen-Reward [18], and on par with the video datasets with pointwise scores. Compared with pointwise scores, we will shortly demonstrate its better alignment with human preference, which is, by definition, a pairwise measurement.

Table 2: Dataset statistics for prior pointwise- and pairwise-supervision resources and our three splits.

Dataset	Cutoff date (MM/YY)	# prompts	# videos	# triplets
Pointwise score				
VideoScore2 [9]	–	2,933	27,168	–
LiFT-HRA [30]	–	–	10k	–
VideoFeedback [10]	–	31.6k	37.6k	–
Pairwise score				
VideoGen-Reward [18]	–	358	4,691	25,233
VPS-AA (ours)	02/26	694	7,372	27,498
VPS-Social (ours)	02/26	694	18,799	269,430
VPS-Combined (ours)	02/26	694	26,171	553,711

4 Learning Automated Video Preference Score

We learn a pairwise preference model, VPS, that predicts which of two videos is more likely to be preferred for a given text prompt. Given a prompt $\boldsymbol{y}$ and two videos $\boldsymbol{x}^A$ and $\boldsymbol{x}^B$, the model outputs $f_\theta(\boldsymbol{y}, \boldsymbol{x}^A, \boldsymbol{x}^B) \in [0, 1]$, which estimates the preference of model A over model B, denoted by $P(A \succ B \mid \boldsymbol{y}, \boldsymbol{x}^A, \boldsymbol{x}^B)$. Unlike pointwise evaluators that assign an absolute quality score to a single video, VPS is trained directly on pairwise comparisons, matching the structure of human preference arenas.

4.1 Preference target construction

Each training example is a tuple $(\boldsymbol{y}, \boldsymbol{x}^A, \boldsymbol{x}^B, q)$, where q is the soft target probability that video A is preferred to video B. The target construction depends on the sources of the two videos:

$$q(A \succ B \mid \boldsymbol{y}, \boldsymbol{x}^A, \boldsymbol{x}^B) = \begin{cases} \sigma\left((R_A - R_B)/100\right), & A, B \in \mathcal{D}_{\text{gen}}, \\ 0.75, & A \in \mathcal{D}_{\text{real}},\ B \in \mathcal{D}_{\text{gen}}, \\ 0.25, & A \in \mathcal{D}_{\text{gen}},\ B \in \mathcal{D}_{\text{real}}, \\ \dfrac{\mathrm{rc}_A}{\mathrm{rc}_A + \mathrm{rc}_B}, & A, B \in \mathcal{D}_{\text{real}}, \end{cases} \tag{1}$$

where $\sigma(z) = 1/(1 + \exp(-z))$ is the sigmoid function following the Bradley-Terry model [2]. For generated–generated pairs, R_A and R_B denote the public Elo scores of the models that generate videos A and B. This converts model-level arena rankings into soft pairwise supervision: a larger Elo gap implies a more confident preference target, while close-performing models produce softer labels.

For real–real pairs, rc_A and rc_B denote the `respond_count` values returned by the social-media retrieval API. The `respond_count` is the total observed engagement with a video, including likes, comments, and other platform-level responses. We use relative response count as the preference signal: video A's winning probability is its response count normalized by the total response count of the pair. Engagement is used *only* for real–real pairs, and *only* as a relative count between two videos retrieved for the same prompt. Engagement remains a noisy preference proxy, as response counts are influenced by creator popularity, trends, and recommendation algorithms. This is different from an Elo-score score because the distribution of the raw response counts is long-tailed, while its logarithm resembles a normal distribution as illustrated in Figure 6. If both response counts are zero, we assign $q = 0.5$.

For mixed real–generated pairs, we assign the social-media video a fixed winning probability of 0.75. This encodes a strong but non-deterministic prior that a relevant real-world video is more likely to be preferred than a generated sample, since it typically contains more natural motion, camera statistics, physical consistency, and visual realism. Under the Bradley–Terry parameterization of Eq. 1, $q = 0.75$ corresponds to an edge of roughly 110 Elo, rather than a deterministic win. We avoid assigning probability 1.0 because retrieval noise and prompt mismatch can make some real-world videos imperfect anchors. We report a sensitivity analysis over q in Appendix D (Table 8). Among the tested values, $q = 0.75$ performs best, while $q = 0.5$, which removes the real-over-generated preference prior, performs below the no-reward baseline.

This target construction is intentionally weakly supervised. Model-level Elo scores do not perfectly determine the quality of every generated sample, and social-media engagement is not a pure measure of perceptual quality. However, both signals are available *at scale*. Our forthcoming experiments test whether a model trained from these noisy pairwise targets, created from AA scraped 694 prompts, recover held-out arena rankings and provide a useful reward signal.

4.2 Preference model

We discuss the design choice of the preference model $f_\theta(\cdot)$.

Metric-only baselines. We first consider a metric-only baseline. This baseline asks whether existing metrics contain enough information to recreate human preference once a learned calibration layer is added. For each video, we compute a vector of existing automatic video-evaluation scores, including text-to-video alignment scores (CLIP [24], SigLIP [39], and InternVideo2 [29]), perceptual and quality metrics (LPIPS [40], DOVER [33], and human ratings (VideoScore [10]). We concatenate the scores with normalization and let $\mathbf{m}^A$ and $\mathbf{m}^B$ denote the metric vectors for videos A and B. We train linear logistic regression and MLP baselines on pairwise metric features.

Proposed end-to-end preference model. Instead of relying on hand-designed metric features, our main model learns the preference score directly from raw pixels with learned video and text representations. We encode each video using Qwen2.5-VL-7B-Instruct and encode the text prompt using `sentence-transformers/all-MiniLM-L6-v2`. For each pair, we concatenate the two video embeddings and the prompt embedding, followed by an MLP g_θ with hidden dimensions of $[8192,\ 4096,\ 2048,\ 1024,\ 512,\ 256,\ 64,\ 4]$.

$$f_\theta(\boldsymbol{y}, \boldsymbol{x}^A, \boldsymbol{x}^B) = \sigma\left(g_\theta\left([\mathbf{e}^A, \mathbf{e}^B, \mathbf{e}^y]\right)\right), \tag{2}$$

where $\mathbf{e}^A, \mathbf{e}^B \in \mathbb{R}^{3584}$ are the video embeddings and $\mathbf{e}^y \in \mathbb{R}^{384}$ is the prompt embedding. The ordering of the two videos is preserved, so the model directly predicts the probability that the first video wins.

Training objective. We train the model with soft binary cross-entropy:

$$\mathcal{L}(\theta) = -q \log f_\theta(\boldsymbol{y}, \boldsymbol{x}^A, \boldsymbol{x}^B) - (1-q) \log\left(1 - f_\theta(\boldsymbol{y}, \boldsymbol{x}^A, \boldsymbol{x}^B)\right). \tag{3}$$

For experiments on the combined dataset, which contains generated–generated, real–real, and real–generated pairs, we upweight AA–AA pairs by a factor of 10 relative to social and cross pairs to prevent the larger splits from overwhelming the smaller but arena-aligned AA split.

$$\mathcal{L}_{\text{weighted}}(\theta) = \frac{\sum_i w_i\, \ell\left(f_\theta(\boldsymbol{y}_i, \boldsymbol{x}_i^A, \boldsymbol{x}_i^B), q_i\right)}{\sum_i w_i}, \qquad w_i = \begin{cases} 10, & \boldsymbol{x}_i^A, \boldsymbol{x}_i^B \in \mathcal{D}_{\text{gen}}, \\ 1, & \text{otherwise.} \end{cases} \tag{4}$$

We train with AdamW using learning rate 10^{-5}, weight decay 10^{-5}, batch size 256, and dropout 0.35. Training runs for up to 1000 epochs with early stopping based on validation loss. We reduce the learning rate on a validation-loss plateau and keep the checkpoint with the best validation loss.

4.3 Evaluation: recreating arena leaderboards

Although VPS is trained on pairwise targets, our main evaluation is whether it can recreate arena-style model rankings. We therefore convert predicted pairwise probabilities into per-model Elo scores and compare the reconstructed leaderboard against the public Artificial Analysis leaderboard.

For a held-out AA–AA pair $(\boldsymbol{x}_i, \boldsymbol{x}_j)$ generated by models m_i and m_j, the preference model predicts

$$\hat{p}_{ij} = f_\theta(\boldsymbol{y}_{ij}, \boldsymbol{x}_i, \boldsymbol{x}_j), \tag{5}$$

where $\hat{p}_{ij}$ is the predicted probability that video i is preferred to video j. We convert this probability into an estimated pairwise Elo gap by applying a scaled logit transform:

$$\widehat{\Delta}_{ij} = K \cdot \text{logit}(\hat{p}_{ij}) = K \cdot \log \frac{\hat{p}_{ij}}{1 - \hat{p}_{ij}}. \tag{6}$$

This gives a signed estimate of the relative quality gap between the two generating models. We then recover per-model scores $\{\hat{R}_m\}$ by solving a least-squares system over all predicted pairwise gaps:

$$\min_{\{\hat{R}_m\}} \sum_{(i,j)} \left[(\hat{R}_{m_i} - \hat{R}_{m_j}) - \widehat{\Delta}_{ij} \right]^2 . \tag{7}$$

Since Elo scores are identifiable only up to an affine transformation, we align the reconstructed scores to the public reference leaderboard with an affine calibration before reporting correlations.

For preference models trained on different splits of the dataset, we first infer the predicted preference score on the AA–AA pairs. This gives a consistent evaluation protocol across all training sources. We evaluate the reconstructed leaderboard using Spearman's ρ, Kendall's τ, top-5 overlap, top-10 overlap, and median absolute rank error against the public leaderboards, including Artificial Analysis and Arena.ai.

4.4 Application: VPS as a reward model

Beyond evaluation, VPS can be used as a plug-in reward model to improve a video generator. Although VPS is trained as a pairwise preference predictor, we convert it into a scalar reward by comparing each generated video against a fixed prompt-matched reference. This yields a stable pointwise reward while preserving the pairwise semantics of the preference model.

We instantiate this reward in Flow-GRPO [17] to fine-tune WAN-2.1 [27]. For each training prompt $\boldsymbol{y}$, we precompute a reference video embedding $\mathbf{e}^{\mathrm{ref}}(\boldsymbol{y})$ from a base WAN-2.1 sample and a prompt embedding $\mathbf{e}^{y}$ from a sentence encoder. During RL training, the current policy generates a video $\boldsymbol{x} \sim \pi_\phi(\cdot \mid \boldsymbol{y})$. We encode $\boldsymbol{x}$ with the same video encoder used by VPS to obtain $\mathbf{e}(\boldsymbol{x})$, and define the reward as

$$r_\theta(\boldsymbol{y}, \boldsymbol{x}) = f_\theta\big(\boldsymbol{y}, \boldsymbol{x}, \boldsymbol{x}^{\mathrm{ref}}(\boldsymbol{y})\big) = \sigma\left(g_\theta\big(\big[\mathbf{e}(\boldsymbol{x}), \mathbf{e}^{\mathrm{ref}}(\boldsymbol{y}), \mathbf{e}^{y}\big]\big)\right), \tag{8}$$

where g_θ denotes the trained MLP preference head in Eq. 2. Intuitively, the reward is high when the generated video is predicted to be preferred over the prompt-matched reference video, and low when it is predicted to lose this comparison. This construction has two practical advantages. First, it avoids running pairwise comparisons against a large reference pool during RL, only one cached reference embedding per prompt is needed. Second, because the reference is fixed for each prompt, the reward remains comparable across multiple samples from the same prompt, which is important for group-relative policy optimization.

The finetuning experiments with WAN-2.1 [27] use Eq. 8 as the only external reward and a KL regularization to prevent the updated policy from drifting too far from the reference generator. The resulting optimization objective is therefore to maximize the VPS preference reward while constraining the updated generator to remain close to the reference policy. This demonstrates that VPS is not only an automatic evaluator for recreating arena rankings, but also a reward signal that can directly improve video generation quality.

5 Experiments and Findings

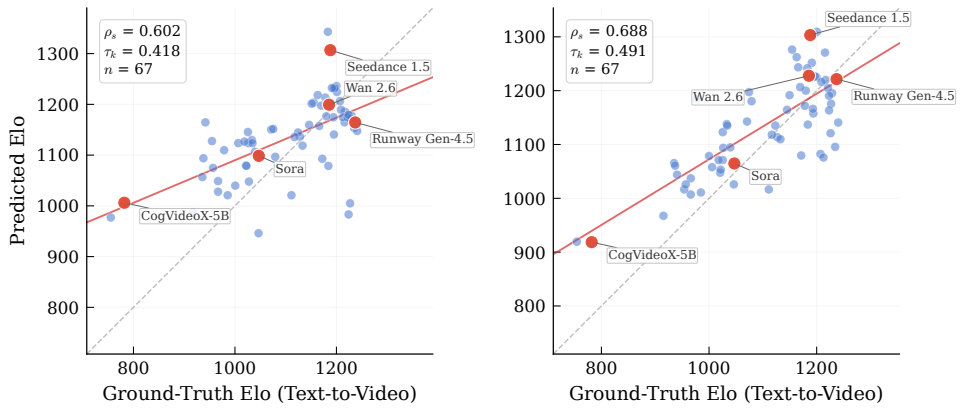
We evaluate VPS along three axes: (1) whether existing automatic metrics are sufficient to recover human preferences, (2) whether the learned preference model can recreate arena leaderboards, and (3) whether the learned score can serve as a reward for improving a video generator.

5.1 Comparison of VPS with Existing metrics

Correlation on model-level human preferences. We first evaluate how much existing automatic metrics align with human preference. We run linear logistic regression and MLP regression to predict pairwise preference and then evaluate by recreating the video arena leaderboard against the public Artificial Analysis (AA) leaderboard. Table 3 shows that existing metrics contain substantial preference signal. The metric MLP outperforms linear

Table 3: Held-out text-to-video leaderboard reconstruction on AA pairs.

Method	Spearman	Kendall τ
Metric-only + Linear	0.590	0.443
VideoScore2 + Linear	0.628	0.456
Metric-only + MLP	0.650	0.465
VPS-AA	**0.688**	**0.491**
VPS-Combined	0.602	0.418



(a) VPS-Combined (b) VPS-AA

Figure 2: Recreated Elo scores versus public Artificial Analysis Elo scores. VPS-AA is specialized for AA-only leaderboard reconstruction, while VPS-Combined learns from both arena pairs and real-world social-video anchors.

regression, indicating that nonlinear combinations of semantic, perceptual, and quality metrics are useful to approximate human arena judgments. However, VPS trained on only AA pairs (VPS-AA) achieves the strongest held-out rank correlation, with Spearman correlation 0.688 and Kendall's $\tau = 0.491$. This suggests that learned prompt-conditioned video representations capture preference information that is not fully represented by fixed metric features. VPS trained on the mixture data (VPS-Combined) slightly underperforms. Figure 2 shows a detailed scatter plot. Despite the reduced correlation, we will shortly show that VPS-Combined's inclusion of real-world videos benefits when VPS is used as a reward model for improving generation.

Agreement with sample-level human preferences. We additionally evaluate VPS-AA zero-shot on MonetBench [36], a human-annotated pairwise preference benchmark. VPS-AA achieves 58.5% pairwise accuracy, competitive with VQAScore (59.5%) and VideoScore (54.9%), both of which are trained on sample-level annotation. This provides sample-level evidence that the learned preference signal transfers to independent human judgments beyond leaderboard reconstruction. Table 4 shows that despite being trained for arena-style holistic preference, VPS-AA transfers to MonetBench's fine-grained, multi-dimensional preference criteria. VisionReward is reported as an in-distribution reference.

Table 4: Pairwise accuracy on MonetBench [36]. VisionReward is evaluated in distribution, while the other methods are evaluated zero-shot.

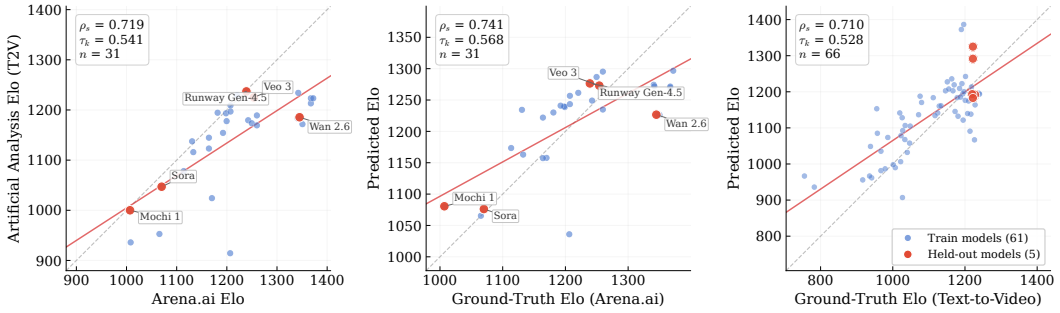
Method	Accuracy (%)
VQAScore [16]	59.5
VideoScore [10]	54.9
VPS-AA	58.5
VisionReward [36]	72.1

5.2 VPS recreates video arena and generalizes across leaderboards and models

VPS generalizes to other leaderboards. To test whether VPS-Combined learns a preference signal beyond the Artificial Analysis leaderboard, we evaluate it on the text-to-video leaderboard from Arena AI[2]. This setting provides an external leaderboard with a different model pool and ranking source. We consider the 31 models that are common to both AA and Arena AI leaderboards. Figure 3 (a-b) shows that VPS-Combined transfers well to this external ranking. On the overlapping model set, the reconstructed leaderboard achieves a Spearman correlation of 0.74 and Kendall's τ of 0.57 with the Arena AI leaderboard. These results suggest that VPS learns a preference signal that generalizes beyond the Artificial Analysis leaderboard.

VPS generalizes to unseen models. A practical evaluator should not only reproduce rankings over models seen during training but also rank newly released models as they appear. To test this, we separate a set of strong text-to-video models from training and use VPS to predict their Elo scores in the reconstructed leaderboard. This simulates a streaming evaluation scenario in which new, improved generators are introduced after the reward model has been trained. Figure 3 (c) shows

[2] `https://arena.ai/leaderboard/text-to-video`



(a) Artificial Analysis vs. Arena.ai (b) Predicted vs. Arena.ai (c) Test — Streaming Model

Figure 3: **(a-b)** Generalization to the Arena AI text-to-video leaderboard. VPS-Combined achieves Spearman correlation 0.74 and Kendall's τ 0.57, indicating that the learned preference model transfers beyond the Artificial Analysis leaderboard. **(c)** Ranking held-out top-tier text-to-video models. Red circles denote strong models excluded from training and evaluated only at test time. These held-out models are still reconstructed near the top of the leaderboard, suggesting that VPS can rank newly introduced high-performing generators without retraining on their arena comparisons.

the reconstructed Elo scores against the original Artificial Analysis Elo scores on the text-to-video test split. The held-out top-tier models are highlighted separately. Despite not being included during training, these models are assigned high reconstructed Elo scores, which is consistent with their ground-truth Elo score. However, it is worth noting that VPS tends to increase the performance gap.

This result is important for practical deployment. Video-generation leaderboards are dynamic: new models are frequently released, and rerunning large-scale human preference studies for every model update is expensive. By accurately placing held-out high-performing models among the top reconstructed ranks, VPS provides evidence that it can support efficient leaderboard updates for newly released video generators.

5.3 VPS provides a useful reward signal for video generation

Finally, we present the effectiveness of VPS as a reward model. The formulation follows Section 4. As reported in Table 5, both learned VPS improve video generation as a reward model. VPS-AA improves WAN-2.1 by $+9$ Elo, while VPS trained with both arena and social videos improves WAN-2.1 by $+51$ Elo. Thus, although adding social-media videos worsens AA-only leaderboard reconstruction, it substantially improves the downstream RL finetuning performance. This suggests that real-world videos provide complementary supervision beyond generated videos.

Table 5: Using VPS as a Flow-GRPO reward improves WAN-2.1. While VPS-AA is more specialized for AA-only leaderboard reconstruction, VPS-Combined provides the stronger reward signal for generator improvement.

Reward model	Elo score	
	Baseline	Flow-GRPO
VPS-AA	937	946 (+9)
VPS-Combined	1066	1117 **(+51)**

Evaluation with VideoScore2 also supports the downstream gains: VPS-Combined achieves 2.683, compared with 2.526 for the base generator, 2.471 with random reward, and 2.577 with VideoScore2 reward (Table 6). Figure 4 shows that the predicted Elo score increases consistently with VPS-Combined while quickly saturating with VPS-AA. We show qualitative comparison in Figure 5 that WAN-2.1 with VPS indeed improves the generation quality in terms of camera motion, content relevance, and object movement. A multi-dimensional VBench evaluation shows improved multiple-object handling but a lower scene score (Table 7), indicating that the improvements vary across quality dimensions. More results are shown in Appendix G.

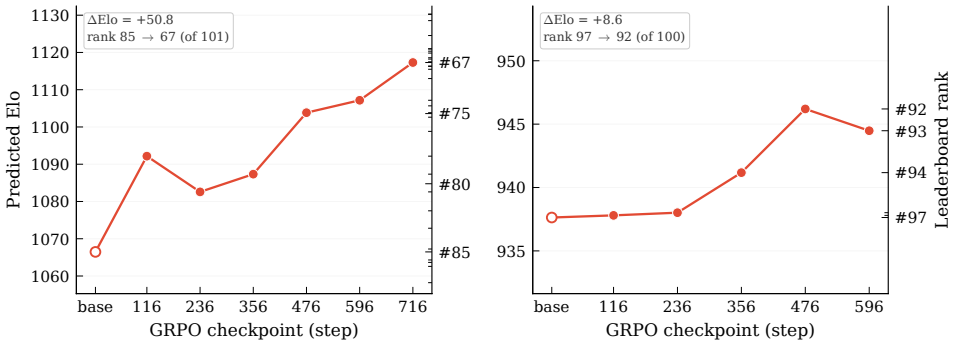
Multi-dimensional generation quality. We evaluate the generator fine-tuned with VPS against the base generator on VBench [12]. Table 7 reports the score changes and their 95% confidence intervals. Multiple-object handling improves, while the scene score decreases. The human-action score increases by 1.33 and appearance style changes decreases by 0.05. These results complement the qualitative motion and alignment improvements while revealing a regression in scene quality.

Table 6: Comparison of different rewards for reinforcement fine-tuning. The resulting generators are evaluated with VideoScore2.

Reward	VideoScore2
No-reward baseline	2.526
Random	2.471
VideoScore2	2.577
VPS-Combined	**2.683**

Table 7: Changes in VBench scores after reinforcement fine-tuning with VPS, relative to the base model. The average covers the four reported dimensions.

Dimension	Change$_{\pm\text{uncertainty}}$
Multiple objects	$+4.37_{\pm 4.32}$
Human action	$+1.33_{\pm 5.34}$
Appearance style	$-0.05_{\pm 0.20}$
Scene	$-3.80_{\pm 2.93}$
Average	$+0.46$



(a) VPS-Combined (b) VPS-AA

Figure 4: VPS-AA vs VPS-Combined as a reward model for video generation.

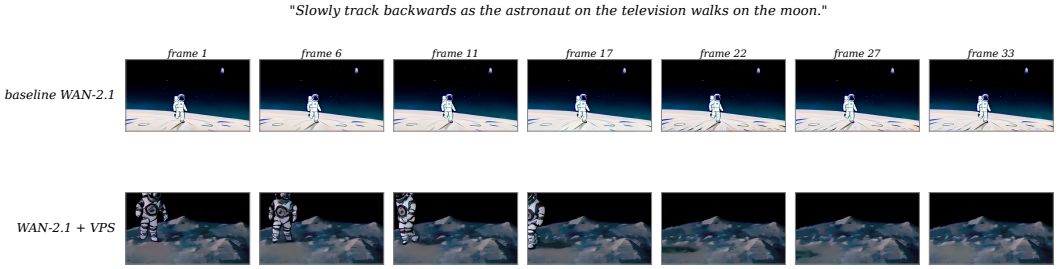


Figure 5: The video generated by WAN-2.1 after reinforcement fine-tuning shows enhanced motion and better alignment with text ("slowly track backward"). We show more results in Appendix G.

6 Conclusions

We propose VPS, a fully automated video preference score learned from weakly supervised signals including model-level Elo scores and user engagement on video-sharing platforms. VPS aligns well with human preference and shows robustness to unseen models and benchmarks. When used as a reward model, VPS greatly improves video generation quality. We hope this work will inspire more accurate and efficient video evaluation protocols.

Acknowledgements. We gratefully acknowledge compute and research support from The Stanford Institute for Human-Centered AI (HAI). This work was supported in part by funding from Google Research. We also thank TarsTech for providing API support.

References

[1] Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.

[2] Ralph Allan Bradley and Milton E Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952.

[3] 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 *ICML*, 2024.

[4] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, 2017.

[5] Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Hao Zhang, Banghua Zhu, Michael Jordan, Joseph E Gonzalez, et al. Chatbot arena: An open platform for evaluating llms by human preference. *arXiv preprint arXiv:2403.04132*, 2024.

[6] Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. In *NeurIPS*, 2017.

[7] Songwei Ge, Aniruddha Mahapatra, Gaurav Parmar, Jun-Yan Zhu, and Jia-Bin Huang. On the content bias in fréchet video distance. In *CVPR*, 2024.

[8] Kaisi Guan, Zhengfeng Lai, Yuchong Sun, Peng Zhang, Wei Liu, Kieran Liu, Meng Cao, and Ruihua Song. Etva: Evaluation of text-to-video alignment via fine-grained question generation and answering. In *ICCV*, 2025.

[9] Xuan He, Dongfu Jiang, Ping Nie, Minghao Liu, Zhengxuan Jiang, Mingyi Su, Wentao Ma, Junru Lin, Chun Ye, Yi Lu, et al. Videoscore2: Think before you score in generative video evaluation. *arXiv preprint arXiv:2509.22799*, 2025.

[10] 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 *EMNLP*, 2024.

[11] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *NeurIPS*, 2017.

[12] 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 *CVPR*, 2024.

[13] Moo Jin Kim, Yihuai Gao, Tsung-Yi Lin, Yen-Chen Lin, Yunhao Ge, Grace Lam, Percy Liang, Shuran Song, Ming-Yu Liu, Chelsea Finn, et al. Cosmos policy: Fine-tuning video models for visuomotor control and planning. *arXiv preprint arXiv:2601.16163*, 2026.

[14] 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 *NeurIPS*, 2023.

[15] Zhi Li, Christos Bampis, Julie Novak, Anne Aaron, Kyle Swanson, Anush Moorthy, and JD Cock. Vmaf: The journey continues. *Netflix Technology Blog*, 25(1), 2018.

[16] Zhiqiu Lin, Deepak Pathak, Baiqi Li, Jiayao Li, Xide Xia, Graham Neubig, Pengchuan Zhang, and Deva Ramanan. Evaluating text-to-visual generation with image-to-text generation. In *ECCV*, 2024.

[17] Jie Liu, Gongye Liu, Jiajun Liang, Yangguang Li, Jiaheng Liu, Xintao Wang, Pengfei Wan, Di Zhang, and Wanli Ouyang. Flow-grpo: Training flow matching models via online rl. In *NeurIPS*, 2025.

[18] 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 *NeurIPS*, 2025.

[19] Yaofang Liu, Xiaodong Cun, Xuebo Liu, Xintao Wang, Yong Zhang, Haoxin Chen, Yang Liu, Tieyong Zeng, Raymond Chan, and Ying Shan. Evalcrafter: Benchmarking and evaluating large video generation models. In *CVPR*, 2024.

[20] Ge Ya Luo, Gian Mario Favero, ZhiHao Luo, Alexia Jolicoeur-Martineau, and Christopher Pal. Beyond fvd: An enhanced evaluation metrics for video generation distribution quality. In *ICLR*, 2025.

[21] Yuhang Ma, Xiaoshi Wu, Keqiang Sun, and Hongsheng Li. Hpsv3: Towards wide-spectrum human preference score. In *ICCV*, 2025.

[22] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *NeurIPS*, 2022.

[23] Jonas Pai, Liam Achenbach, Victoriano Montesinos, Benedek Forrai, Oier Mees, and Elvis Nava. mimic-video: Video-action models for generalizable robot control beyond vlas. In *RSS*, 2026.

[24] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

[25] Jingqi Tong, Yurong Mou, Hangcheng Li, Mingzhe Li, Yongzhuo Yang, Ming Zhang, Qiguang Chen, Tianyi Liang, Xiaomeng Hu, Yining Zheng, et al. Thinking with video: Video generation as a promising multimodal reasoning paradigm. *arXiv preprint arXiv:2511.04570*, 2025.

[26] Thomas Unterthiner, Sjoerd Van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric & challenges. *arXiv preprint arXiv:1812.01717*, 2018.

[27] Team Wan, 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.

[28] 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. *arXiv preprint arXiv:2307.06942*, 2023.

[29] Yi Wang, Kunchang Li, Xinhao Li, Jiashuo Yu, Yinan He, Guo Chen, Baoqi Pei, Rongkun Zheng, Zun Wang, Yansong Shi, et al. Internvideo2: Scaling foundation models for multimodal video understanding. In *ECCV*, 2024.

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

[31] Zhou Wang, Ligang Lu, and Alan C Bovik. Video quality assessment based on structural distortion measurement. *Signal processing: Image communication*, 19(2):121–132, 2004.

[32] Thaddäus Wiedemer, Yuxuan Li, Paul Vicol, Shixiang Shane Gu, Nick Matarese, Kevin Swersky, Been Kim, Priyank Jaini, and Robert Geirhos. Video models are zero-shot learners and reasoners. *arXiv preprint arXiv:2509.20328*, 2025.

[33] Haoning Wu, Erli Zhang, Liang Liao, Chaofeng Chen, Jingwen Hou, Annan Wang, Wenxiu Sun, Qiong Yan, and Weisi Lin. Exploring video quality assessment on user generated contents from aesthetic and technical perspectives. In *ICCV*, 2023.

[34] Jay Zhangjie Wu, Guian Fang, Haoning Wu, Xintao Wang, Yixiao Ge, Xiaodong Cun, David Junhao Zhang, Jia-Wei Liu, Yuchao Gu, Rui Zhao, et al. Towards a better metric for text-to-video generation. *arXiv preprint arXiv:2401.07781*, 2024.

[35] Xiaoshi Wu, Keqiang Sun, Feng Zhu, Rui Zhao, and Hongsheng Li. Human preference score: Better aligning text-to-image models with human preference. In *ICCV*, 2023.

[36] Jiazheng Xu, Yu Huang, Jiale Cheng, Yuanming Yang, Jiajun Xu, Yuan Wang, Wenbo Duan, Shen Yang, Qunlin Jin, Shurun Li, et al. Visionreward: Fine-grained multi-dimensional human preference learning for image and video generation. In *AAAI*, 2026.

[37] Mengjiao Yang, Yilun Du, Kamyar Ghasemipour, Jonathan Tompson, Dale Schuurmans, and Pieter Abbeel. Learning interactive real-world simulators. In *ICLR*, 2024.

[38] Seonghyeon Ye, Yunhao Ge, Kaiyuan Zheng, Shenyuan Gao, Sihyun Yu, George Kurian, Suneel Indupuru, You Liang Tan, Chuning Zhu, Jiannan Xiang, et al. World action models are zero-shot policies. *arXiv preprint arXiv:2602.15922*, 2026.

[39] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *ICCV*, 2023.

[40] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018.

[41] Sixian Zhang, Bohan Wang, Junqiang Wu, Yan Li, Tingting Gao, Di Zhang, and Zhongyuan Wang. Learning multi-dimensional human preference for text-to-image generation. In *CVPR*, 2024.

[42] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *NeurIPS*, 2023.

A Limitations

We discuss potential limitations as follows.

Weak supervision and imperfect proxies. Both Elo scores at the model level from the video leaderboard and user engagements from the social media provide only noisy signals. They do not perfectly capture intrinsic perceptual quality. Therefore, the results should be interpreted as an empirical finding rather than a claim that social engagement alone is sufficient for true human preference alignment. The proposed VPS serves as a computationally efficient alternative.

Evaluation precision. Although VPS retains global arena rankings with high correlation, it demonstrates imperfect reconstruction at the very top of the leaderboards, reflected in a lower top-K overlap metric. This indicates that recovering the exact order of top-performing models is more challenging than recovering the global trend. This is also probably due to the relatively small gap between the top-performing models nowadays.

Social biases. User engagement can be skewed due to other factors, such as trends, individual influencers, and recommendation algorithms, all of which could potentially result in social biases.

B Societal Impacts

B.1 Potential Positive Societal Impacts

Democratized Evaluation. By replacing costly, slow, and hard-to-replicate human rating arenas with a fully automated, highly correlated metric, VPS significantly lowers the barrier to entry for evaluating video generation models. This allows smaller academic labs and independent researchers to rapidly test and iterate on their models without the need for large-scale human annotation.

Improved Model Alignment. As demonstrated by its use as a reward model with Flow-GRPO, VPS provides a scalable feedback mechanism to improve the visual quality and prompt alignment of generative videos. This leads to more capable tools for creative professionals, educators, and content creators who rely on text-to-video technologies.

Reproducibility in Research. Human evaluations are notoriously sensitive to participant demographics, fatigue, and interface design. VPS provides a deterministic, computational framework for leaderboard recreation, fostering better reproducibility and standardized benchmarking within the generative AI community.

B.2 Potential Negative Societal Impacts & Risks

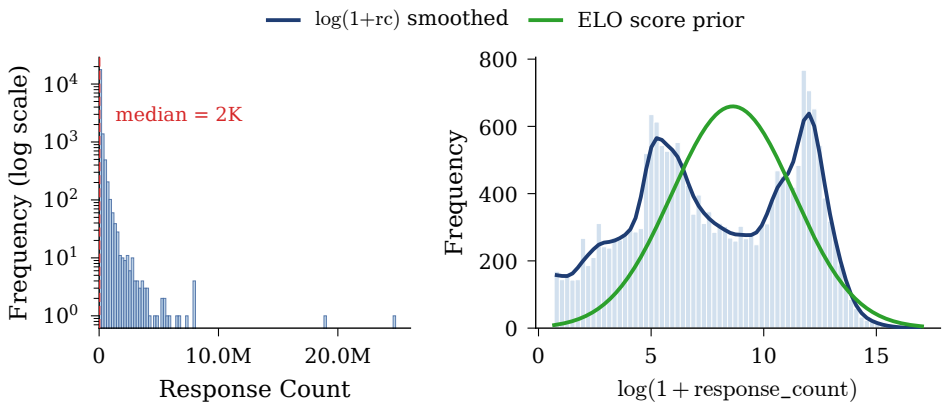
Amplified Social Media Bias. A core innovation of VPS is the utilization of social media engagement (e.g., likes, comments, and response counts) as a proxy for human preference. However, social media engagement is heavily influenced by algorithmic recommendation biases. Training a reward model on this data risks implicitly teaching video generators to prefer sensationalist or biased visual styles over objective realism or diverse representation.

C Safeguards

We implemented automated safety filters to remove NSFW or highly graphic content from the scraped social media videos prior to inclusion in the dataset. Furthermore, the dataset and pre-trained VPS reward model will be gated behind a Data Use Agreement (DUA) that requires researchers to verify their academic affiliation and agree not to use the dataset and the pre-trained reward model to optimize generators for explicit, deceptive, or non-consensual synthetic media.

D Additional Evaluation

Sensitivity to the real–generated prior. We sweep the real-over-generated preference prior over $q \in \{0.5, 0.75, 0.85\}$ and evaluate each resulting reward's effect on the fine-tuned generator with VideoScore2. Table 8 shows that $q = 0.75$ performs best among the tested values, reaching 2.664. The stronger prior $q = 0.85$ also improves over the no-reward baseline, whereas $q = 0.5$, which



(a) Raw Distribution (b) Log-Transformed Distribution

Figure 6: Respond count distribution

removes the real-over-generated preference prior, falls below it. These results support $q = 0.75$ as an empirical operating point.

Table 8: Sensitivity to the real-over-generated preference prior, evaluated through the resulting fine-tuned generator using VideoScore2.

Prior	VideoScore2
No-reward baseline	2.526
$q = 0.50$	2.465
$q = 0.75$	**2.664**
$q = 0.85$	2.577

E Distribution of User Engagements

In Figure 6, we show that the raw response counts distribution is in a long-tailed distribution. After we take the logarithm, the distribution resemble a normal distribution, which is the Elo score prior, more closely.

F Example Training Pairs

Figure 7 illustrates how the training pairs for VPS are created.

G More Qualitative Results

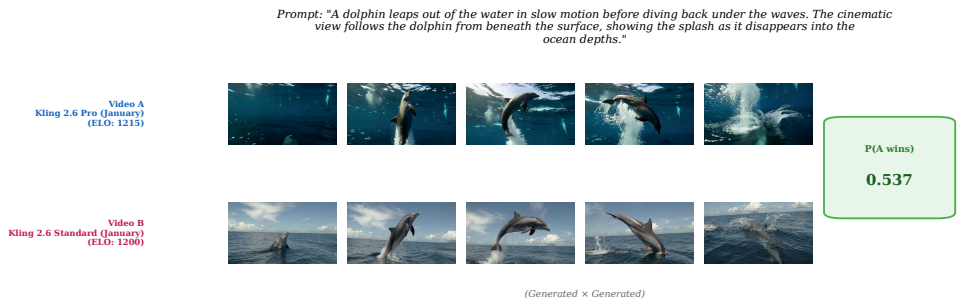
Figure 8 illustrates some examples of how VPS improves the quality of videos produced by WAN2.1.

H Compute Resource

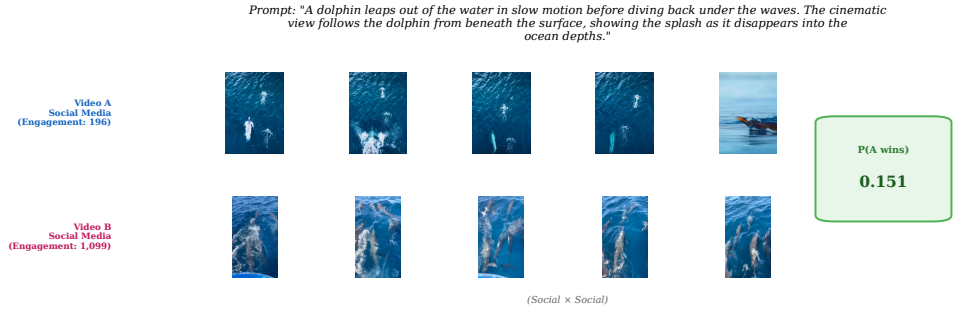
The preference model is trained on an internal cluster with NVIDIA L40s GPUs and takes around 15 minutes to train.

The Flow-GRPO experiments are conducted on an internal cluster with NVIDIA H100 GPUs due to a larger GPU memory requirement. Each experiment runs on $2\times$ H100 GPUs for 3 days, amounting to ~ 150 GPU-hours per job.

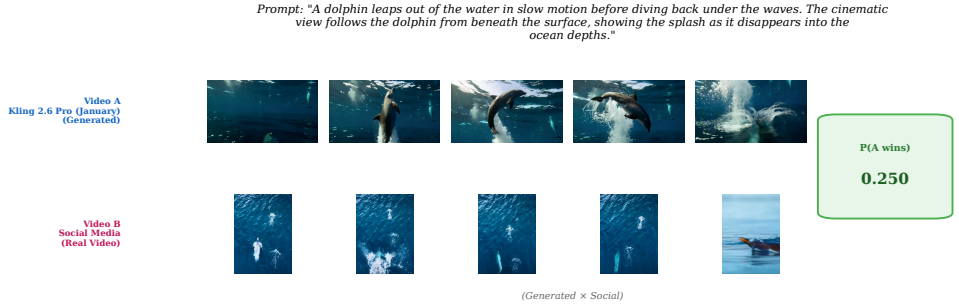
Preliminary and failed experiments do not take a significant amount of computation.



(a) training pair from two generated videos



(b) training pair from two real world social media videos

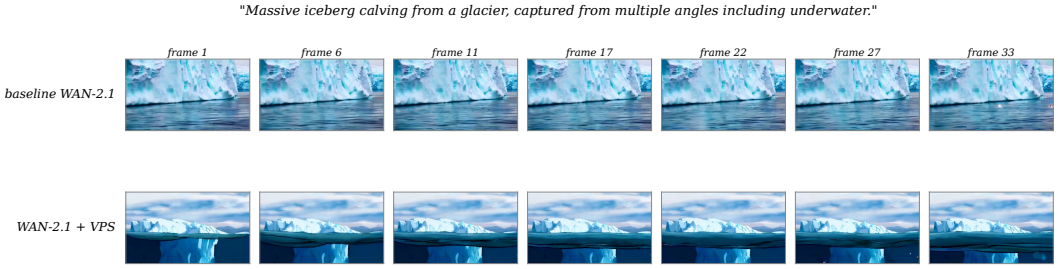


(c) training pair from a generated and a social media video

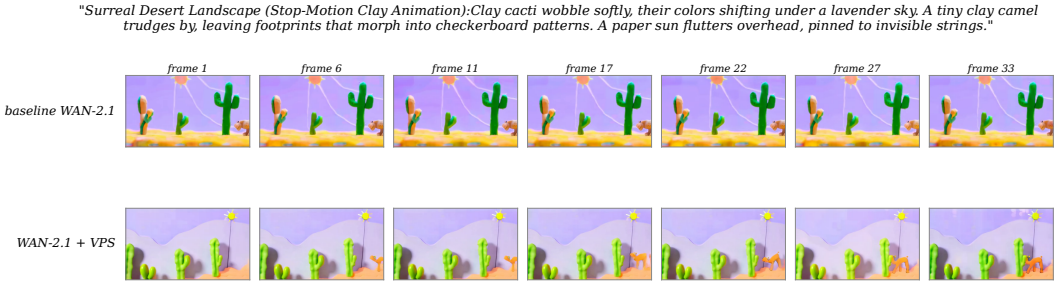
Figure 7: Example training pairs with assigned preference score

I Licenses for existing assets

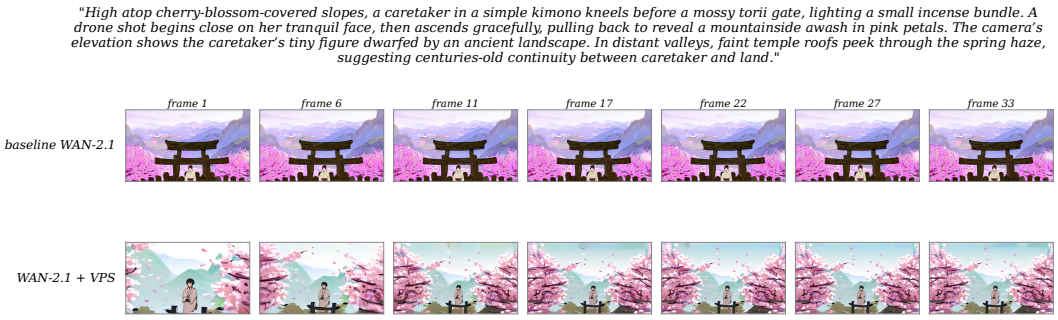
FlowGRPO - MIT.
VBench - Apache 2.0
Videoscore2 - MIT
Dover - S-Lab License 1.0
LPIPS - BSD-2-Clause
SigLIP - Apache 2.0
InternVideo2 - Apache 2.0



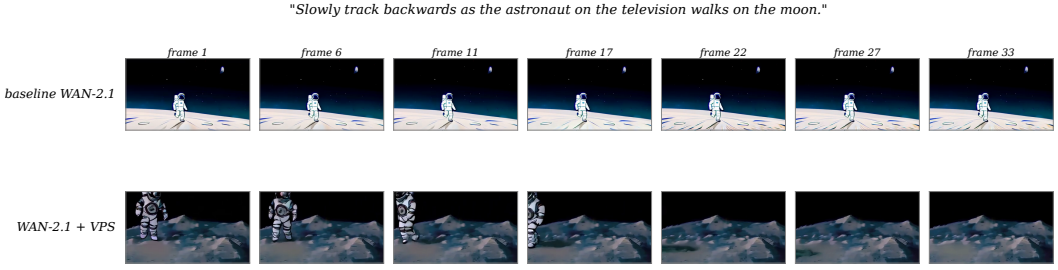
(a) baseline model fails to capture the iceberg underwater



(b) baseline model fails to capture the movement of the camel



(c) As per the prompt the camera is supposed to zoom out, but baseline model fails to capture this



(d) baseline model fails to capture the movement of the astronaut

Figure 8: VPS improves the alignment and hence the human preference over videos generated by WAN-2.1.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: The abstract and introduction accurately describe the paper's key contributions in terms of data collection (details in Sec. 3), learned video preference score (Sec. 4), and key findings (Sec. 5).

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: We have a separate "Limitations" section in Sec. A concerning the imperfection of the proposed evaluation metric and potential social biases from the data.

   Guidelines:

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

3. **Theory assumptions and proofs**

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

Answer: [N/A]

Justification: The paper does not include theoretical results.

Guidelines:

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

4. **Experimental result reproducibility**

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

Answer: [Yes]

Justification: The paper fully discloses all the information needed to reproduce the main experimental results of the paper. The code and the data are provided. At the time of submission, they are uploaded to `https://huggingface.co/datasets/supervideoVPS/super_video/tree/main` and `https://anonymous.4open.science/r/super_flow_grpo`.

Guidelines:

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

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

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

   Answer: [Yes]

   Justification: We provide open access code and the data, with sufficient instructions to reproduce the main experimental results. At the time of submission, they are uploaded to `https://huggingface.co/datasets/supervideoVPS/super_video/tree/main` and `https://anonymous.4open.science/r/super_flow_grpo`.

   Guidelines:

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

6. **Experimental setting/details**

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

   Answer: [Yes]

   Justification: We discuss the data splits in Sec. 3.4 and the training details in Sec. 4.2.

   Guidelines:

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

7. **Experiment statistical significance**

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

   Answer: [No]

   Justification: The paper does not report error bars.

   Guidelines:

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

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

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: The compute resources are summarized in Sec. H.

   Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: The research condcuted in the paper conform, in every respect, with the NeurIPS Code of Ethics.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [Yes]

    Justification: We discuss both potential positive societal impacts and negative societal impacts in Sec. B.

    Guidelines:

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

11. **Safeguards**

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

    Answer: [Yes]

    Justification: We describe safeguards for responsible data and model release in Sec. C.

    Guidelines:

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

12. **Licenses for existing assets**

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

    Answer: [Yes]

    Justification: The creators and original owners of assets used in the paper are properly credited. The license and terms of use, summarized in Sec. I, are properly respected.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.

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

13. **New assets**

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

    Answer: [Yes]

    Justification: The new assets, include the data and code, are well documented. Please refer to `https://huggingface.co/datasets/supervideoVPS/super_video/tree/main` and `https://anonymous.4open.science/r/super_flow_grpo`.

    Guidelines:

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

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

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

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

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

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

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

    Answer: [N/A]

    Justification: We consulted the school office in charge of IRB approvals and were informed that our paper required no IRB review. Below is their response:

```
...
Based on the information provided in the protocol, this research
does not involve human subjects as defined in 45 CFR 46.102(e)(1).
As such, no IRB review is required for this project. We will
withdraw your protocol and you are free to proceed with this project
without further IRB review.
```

Guidelines:

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

16. **Declaration of LLM usage**

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

Answer: [N/A]

Justification: The paper does not require LLMs as a component of the core methods in this research. The LLM is only used for polishing the language of the manuscript. It does not impact the core methodology, scientific rigor, or originality of the research.

Guidelines:

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