Behavior Pack Optimization for Video MLLM Post-Training

Zhaolu Kang[1,2*], **Shiyu Liu**[1,*],
Tailong Luo, **Wei Zhang**, **Yingjie He**[1], **Guangyuan Dong**, **Lei Wei**[1], **Guansu Wang**[3],
Liang He, **Siheng Wang**, **Jiaqi Su**, **Shuang Chen**, **Haoyu Ji**, **Qishi Zhan**, **Kaiyue Zhou**[2,†]
[1]Peking University [2]Chengdu Minto Tech [3]The University of Melbourne
zlkang25@stu.pku.edu.cn kyzhou@wayne.edu

Abstract

Video multimodal large language models (MLLMs) keep climbing video question answering benchmarks, yet shuffling the frames, masking the segment that supports the answer, or occluding the target object barely changes their predictions. The accuracy rests on appearance and language priors, not on the temporal evidence the question asks for. We trace this to the unit of post-training: rewards are computed on a single response to the original clip, so the model is never asked to behave consistently across views. We propose Behavior Pack Optimization (BPO), which replaces the single response with a behavior pack of outputs across counterfactual views chosen by question type, scored jointly. The pack reward asks for stability when the intervention is irrelevant, sensitivity when key evidence is removed, and abstention when no evidence remains. To keep this objective stable at small pack sizes, BPO uses an anchor-relative advantage: the response on the original view serves as a per-prompt reference instead of a group mean over mixed views. On TempCompass, MVBench, and NExT-QA, BPO improves the macro accuracy of Qwen2.5-VL-7B-Instruct by 4.7 pp, the temporal-hard subset by 7.8 pp, and abstention F1 by 20.0 pp over a budget-matched vanilla GRPO baseline from the same SFT checkpoint. The gains transfer to Video-MME, LongVideoBench, and to LLaVA-Video-7B; ablations confirm they follow the view sets, not the rollout count. We hope this pack-level perspective offers a useful starting point for the video MLLM and multimodal post-training community as the field moves toward evidence-grounded video reasoning.

1 Introduction

Video multimodal large language models (MLLMs) are now common backbones for video question answering. Their post-training recipe, however, still often follows the text-only LLM pipeline: supervised fine-tuning on instruction-formatted question-answer pairs, optionally followed by preference optimization or group-relative reinforcement learning [1, 2, 3]. Recent video RL methods add temporal rewards, augmented negatives, or pairwise preferences over manipulated clips [4, 5]. These signals are useful, but the optimized object often remains local: one response on one video, a same-input response group, or a comparison tied to one edited clip. This design fits static answer matching, but it misses a defining property of video QA: the answer should often change, or become unsupported, when the visual evidence is changed or removed.

A local optimization unit creates a structural gap. When the loss grades responses in isolation, the policy receives no direct signal about whether its behavior changes appropriately across related counterfactual views. Benchmarks such as MVBench and TempCompass stress temporal order, direction, and evidence-dependent reasoning [6, 7], while controlled probes show that video MLLMs

*Equal Contribution. †Corresponding Author.

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

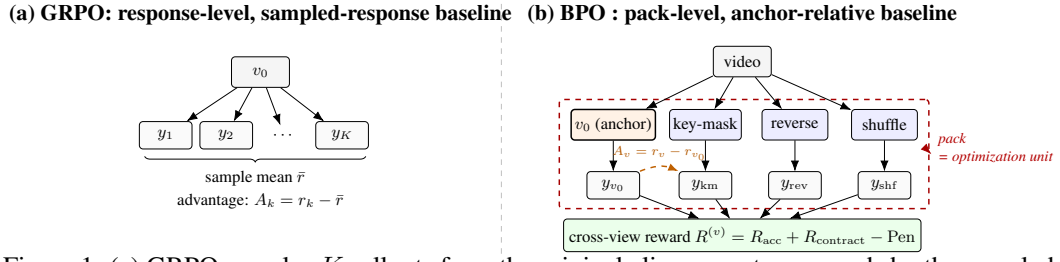

Figure 1: (a) GRPO samples K rollouts from the original clip v_0, centers rewards by the sampled-response mean $\bar{r}$, and optimizes each response separately. (b) BPO samples one rollout per view from a question-type-conditioned set, scores the pack jointly with cross-view behavior contracts, and centers advantages relative to the v_0 anchor rollout.

can preserve answers under frame shuffling, object occlusion, or removal of answer-supporting segments. Adding manipulated clips broadens the training distribution, but if each response is still graded against a local target, the objective does not specify how the model should behave across interventions. The bottleneck is therefore not only what the model is shown, but also the granularity at which behavior is optimized.

We propose **B**ehavior **P**ack **O**ptimization (BPO), a video RL post-training framework that optimizes a behavior pack instead of an isolated response. For each video-question pair, BPO constructs a compact pack with the original clip and question-type-conditioned counterfactual views: evidence-removal views for event, binding, and count questions, and temporal interventions such as reversal or shuffling for order and direction questions. The policy is rolled out on every view, and the outputs are scored jointly by a verifier-based reward that combines per-view correctness with three cross-view contracts: stability under label-preserving perturbations, sensitivity to answer-changing interventions, and honest abstention when no diagnostic evidence remains. Thus, BPO is not simply data augmentation: it changes the optimization unit from a single video-answer pair to a cross-view behavior pack.

Within this objective, the advantage estimator also matters. Standard GRPO centers rewards by statistics of the sampled group, but video post-training often permits only small groups because of memory and decoding cost. BPO instead uses the rollout on the original, unintervened clip as a per-prompt anchor, yielding an anchor-relative advantage across the pack. The anchor is graded against the ground-truth answer on the original clip, so intervened-view rewards alone cannot set the reference. We analyze this estimator in Section 3.3 and show that it can reduce variance when rewards across the anchor and intervened views are correlated.

We evaluate BPO on TempCompass, MVBench, and NExT-QA [8] using Qwen2.5-VL-7B-Instruct as the backbone. All main comparisons use the same SFT initialization and match the per-prompt rollout budget under compute-controlled training. Against vanilla GRPO and budget-matched re-implementations of Video-R1 [4] and TPO [5], BPO improves macro accuracy by 4.7 pp, temporal-hard accuracy by 7.8 pp, and abstention F1 on insufficient-evidence subsets by 20.0 pp. The behavior indicators move in the directions specified by the contracts; ablations separate type-aware interventions from extra rollouts; and the gains transfer to longer-video evaluation and a second backbone.

Our contributions are:

- We identify optimization granularity, rather than data coverage alone, as a bottleneck connecting shortcut behavior in video QA to the post-training objective.
- We introduce BPO, a pack-level objective that combines question-type-conditioned view sets, cross-view behavior contracts, and an anchor-relative advantage estimator.
- Through budget-aligned comparisons, we show that BPO improves accuracy, abstention, and contract-aligned behavior indicators, with gains transferring to longer-video evaluation and a second backbone.

2 Related Work

Faithful temporal grounding. Video MLLMs are commonly post-trained with supervised fine-tuning on instruction-style video question-answer pairs [9, 10, 11], and evaluated on benchmarks that stress temporal order, multi-step events, and long-video reasoning [6, 7, 8, 12, 13]. High

benchmark scores, however, do not guarantee faithful temporal grounding. A model may answer from appearance cues or language priors, or preserve the same answer after the supporting evidence is reordered, masked, or removed. Counterfactual evaluation and evidence-sensitivity diagnostics therefore complement static answer matching: a faithful model should remain stable under label-preserving perturbations, but change or abstain when an intervention changes or removes the evidence needed to answer [5, 14].

Video RL and counterfactual supervision. Preference optimization and reinforcement learning offer a way to improve behavior beyond supervised imitation. DPO learns from preference pairs without an explicit reward model, while GRPO removes the value model and estimates relative advantages from responses sampled for the same prompt [1, 2, 3]. Recent video post-training methods introduce video-specific rewards and training signals: Video-R1 uses shuffled-frame temporal rewards, VideoChat-R1 applies GRPO-style RFT to spatio-temporal perception, VideoRFT adds semantic-consistency rewards, DeepVideo-R1 studies difficulty-aware GRPO variants, and SpaceR focuses on spatial RLVR [4, 15, 16, 17, 18]. Another line constructs preferences from edited, temporally perturbed, or synthesized counterfactual videos [5, 14]. These works broaden the supervision available to video post-training, but the training signal is typically applied to an individual response, a same-input response group, or a pairwise comparison tied to one manipulated instance, rather than to the joint behavior across a structured set of views.

Behavior-pack optimization. BPO changes this optimization granularity. It optimizes a compact question-type-conditioned behavior pack in which outputs from the original clip and diagnostic counterfactual views are scored jointly under contracts for stability, sensitivity, and honest abstention. This is orthogonal to many reward or preference designs: BPO keeps the GRPO-style clipped surrogate, but changes the unit being optimized and replaces the batch-normalized pack reference with an anchor-relative reference from the original-clip rollout. It is also distinct from reasoning-trace or representation-based approaches, which improve how models structure intermediate reasoning [19]; BPO changes the unit over which video behavior is rewarded.

3 Method

We cast video RL post-training as cross-view behavior optimization. For each video-question sample, Behavior Pack Optimization (BPO) builds the original view and a small set of question-typed counterfactual views. The policy produces one rollout per view; a verifier scores view-level correctness and cross-view contracts relative to the original-view anchor.

Compared with vanilla GRPO, BPO changes both the optimization unit and the baseline: it trains on behavior packs rather than isolated responses, and it uses an anchor-relative advantage rather than a batch-normalized pack baseline. This makes the training signal depend on whether answers change, stay stable, or abstain under evidence interventions. Figure 2 summarizes the pipeline; the following subsections define its components.

3.1 Setup and view records

A training sample is $(\mathbf{V}, q, y^\star)$, optionally with structured metadata such as key-evidence segments, target objects, and counterfactual annotations. The policy π_θ outputs an answer and an Entity-State-Temporal (EST) trace installed during Stage A SFT (Appendix B.2); a deterministic grammar parses the trace with success rate ≥ 0.99.

We represent each view as a record

$$\omega_v = (v, \mathbf{V}_v, q_v, y_v^\star, \xi_v), \tag{1}$$

where v denotes the intervention type, $\mathbf{V}_v$ is the transformed video, q_v is the corresponding question, $y_v^\star$ is the view-specific target answer, and ξ_v stores intervention metadata used by the verifier and the contract reward. The identity view v_0 satisfies

$$(\mathbf{V}_{v_0}, q_{v_0}, y_{v_0}^\star) = (\mathbf{V}, q, y^\star). \tag{2}$$

For counterfactual views, $y_v^\star$ is inherited for label-preserving views, derived from deterministic counterfactual rules when available, or replaced by an insufficient-evidence target when the intervention removes the required support. The full view pool is

$$\mathcal{V}_{\text{full}} = \{v_0, \text{key-mask}, \text{non-key-mask}, \text{span-drop}, \text{reverse}, \text{shuffle}\}. \tag{3}$$

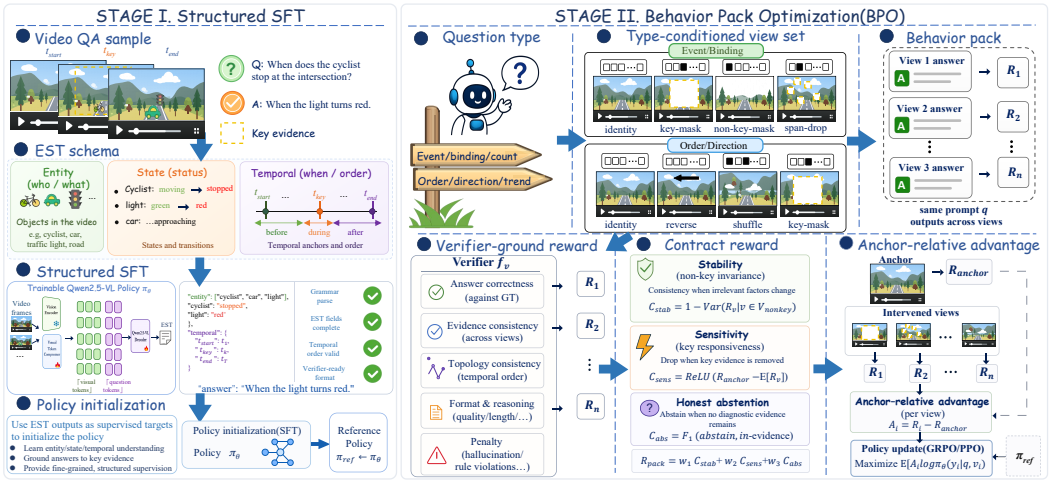

Figure 2: Overview of Behavior Pack Optimization. Stage A installs the structured EST interface with SFT. Stage B routes each question to a question-typed view set, forms a behavior pack across counterfactual views, scores view-specific and cross-view contract rewards, and updates the policy with anchor-relative advantages.

View construction details are in Appendix B.3.

3.2 Behavior packs

For a sample $(\mathbf{V}, q)$ and a view set $\mathcal{V} \subseteq \mathcal{V}_{\text{full}}$ with $v_0 \in \mathcal{V}$, the policy produces a behavior pack

$$\mathcal{B}(\mathbf{V}, q; \mathcal{V}) = \Big\{(v, y_v) : v \in \mathcal{V},\ \omega_v = (v, \mathbf{V}_v, q_v, y_v^\star, \xi_v),\ y_v \sim \pi_\theta(\cdot \mid \mathbf{V}_v, q_v)\Big\}. \tag{4}$$

Unlike GRPO response groups, BPO samples across views of the same prompt, so a pack exposes whether the policy preserves, changes, or abstains as evidence is intervened on. This is different from treating counterfactual clips as ordinary data augmentation: the views are decoded in the same optimization unit and are scored jointly through contracts to the original view. As a result, the update can reward a response that stays unchanged under a label-preserving edit while penalizing the same unchanged answer after evidence has been removed.

Pack-level batch baseline. For the ablation baseline, the scalar pack reward is the mean per-view reward:

$$R_{\text{pack}}(\mathcal{B}) = \frac{1}{|\mathcal{V}|} \sum_{(v, y_v) \in \mathcal{B}} R_{\text{view}}(v, y_v; y_{v_0}). \tag{5}$$

For a mini-batch of G packs, the batch-normalized pack advantage is

$$\hat{A}^{(i)}_{\text{pack}} = \frac{R_{\text{pack}}\big(\mathcal{B}^{(i)}\big) - \mu_G}{\sigma_G + \epsilon}, \qquad \mu_G, \sigma_G \text{ computed over the } G \text{ packs in the mini-batch.} \tag{6}$$

The clipped surrogate applies this shared pack-level advantage view-wise:

$$\mathcal{L}_{\text{group}} = -\mathbb{E}_{\mathcal{B}^{(i)}}\left[\frac{1}{|\mathcal{V}|} \sum_{(v, y_v) \in \mathcal{B}^{(i)}} \min\Big(\rho_v \hat{A}^{(i)}_{\text{pack}}, \text{clip}(\rho_v, 1-\epsilon, 1+\epsilon)\hat{A}^{(i)}_{\text{pack}}\Big)\right], \tag{7}$$

where

$$\rho_v = \frac{\pi_\theta(y_v \mid \mathbf{V}_v, q_v)}{\pi_{\text{old}}(y_v \mid \mathbf{V}_v, q_v)} \tag{8}$$

is the per-view importance ratio. With only the identity view and no cross-view contracts, this reduces to a single-view clipped update. With multiple views, the shared pack reward couples responses through the same intervention set.

Small-pack estimation. Since video RL permits only small packs ($|\mathcal{V}| \leq 4$ and $G \approx 8$–16), batch centering in Equation (6) mixes prompt difficulty with intervention behavior and can yield noisy variance estimates. The normalization is over packs in the mini-batch, not over multiple rollouts of the same prompt as in standard GRPO. BPO replaces this batch-level reference with a prompt-level anchor.

3.3 Anchor-relative optimization

BPO uses the original-view rollout as a prompt-specific reference for all intervened views, avoiding reward comparisons across unrelated prompts.

Definition 1 (Anchor-relative advantage)**.** *For pack $\mathcal{B}^{(i)} = \{(v, y_v^{(i)})\}_{v \in \mathcal{V}}$ with anchor view v_0, the anchor-relative advantage of view v is*

$$\hat{A}_{\text{anc}}^{(i,v)} = R_{\text{view}}\big(v, y_v^{(i)}; y_{v_0}^{(i)}\big) - R_{\text{view}}\big(v_0, y_{v_0}^{(i)}; y_{v_0}^{(i)}\big). \tag{9}$$

The anchor view has zero self-relative advantage, $\hat{A}_{\text{anc}}^{(i,v_0)} = 0$.

The BPO surrogate applies the clipped policy-gradient objective to the non-anchor views:

$$\mathcal{L}_{\text{BPO}} = -\mathbb{E}_i \left[\frac{1}{|\mathcal{V}|} \sum_{(v,y_v) \in \mathcal{I}^{(i)}} \min(\rho_v A_v, c_v A_v) \right] + \beta_{\text{KL}}\, \mathbb{E}_{v \in \mathcal{V}} \text{KL}[p_{\theta,v} \| p_{\text{ref},v}]\,. \tag{10}$$

where $\mathcal{I}^{(i)} = \{(v, y_v) \in \mathcal{B}^{(i)} : v \neq v_0\}$, $A_v = \hat{A}_{\text{anc}}^{(i,v)}$, $c_v = \text{clip}(\rho_v, 1-\epsilon, 1+\epsilon)$, $p_{\theta,v} = \pi_\theta(\cdot \mid \mathbf{V}_v, q_v)$, and $p_{\text{ref},v} = \pi_{\text{ref}}(\cdot \mid \mathbf{V}_v, q_v)$. The normalization remains pack-level; the anchor has zero relative advantage and acts as the grounded control for all non-anchor updates, while the KL term keeps Stage B close to the Stage A reference policy. We do not use the anchor-relative term to further optimize the original-view answer itself: otherwise a high- or low-reward anchor could move the reference it is supposed to provide. The original view contributes through the KL term and through its reward as the baseline used to form non-anchor advantages. We log its verifier score in the pack diagnostics, but do not apply an additional RL log-probability term to the anchor response.

Estimator stability and budget. Under the exchangeable reward model in Appendix A.2, the anchor-relative update signal has lower unnormalized variance than the batch-centered pack signal when $\rho > \rho^\star(m) = (2m-1)/(3m-1)$. The measured correlations in Section 4.3 exceed this threshold. We also match the total rollout budget to vanilla GRPO by choosing group sizes with $|\mathcal{V}| G_{\text{BPO}} \approx K_{\text{vanilla}} G_{\text{vanilla}}$ within $\pm 5\%$; measured FLOPs are reported in Appendix B.8.

3.4 Question-typed counterfactual views

BPO matches interventions to the evidence demanded by the question: event and binding questions require evidence-presence probes, while order and direction questions require temporal probes.

- **Type 1: event, binding, and count.** BPO uses

$$\mathcal{V}_1 = \{v_0, \text{key-mask}, \text{non-key-mask}, \text{span-drop}\}, \tag{11}$$

  which probes whether the answer depends on the relevant evidence rather than background context.

- **Type 2: order, direction, and trend.** BPO uses

$$\mathcal{V}_2 = \{v_0, \text{reverse}, \text{shuffle}, \text{key-mask}\}, \tag{12}$$

  which probes temporal dependence through order and witness interventions.

A lightweight rule classifier maps $q \mapsto \tau(q) \in \{1, 2\}$ using interrogative templates and verb frames. Details of the classifier and the oracle-versus-predicted comparison are given in Appendix B.6. The diagonal experiment in Section 4.4 verifies that the matched pairing between question type and view family matters: swapping the view sets across question types degrades temporal-hard accuracy, indicating that the gain is not simply from adding extra views.

3.5 Verifier-grounded contract reward

For each view v, the verifier produces a scalar reward $R_{\text{view}}(v, y_v; y_{v_0})$. The reward combines correctness within the current view, structured evidence checks, cross-view behavior contracts, and penalties for degenerate outputs:

$$R_{\text{view}}(v, y_v; y_{v_0}) = w_1 R^{(v)}_{\text{ans}}(y_v; y^\star_v) + w_2 R_{\text{ev}}(v, y_v) + w_3 R_{\text{topo}}(v, y_v) + w_4 R_{\text{fmt}}(y_v) + \lambda_v R^{(v)}_{\text{contract}}(y_v, y_{v_0}) - \beta_v \text{Pen}_v(y_v, \mathcal{B}). \tag{13}$$

The first four terms score current-view answer correctness, evidence support, EST consistency, and output format. The contract term scores the relation to the anchor response; for v_0 we set $\lambda_{v_0} = 0$, so the anchor remains grounded in the original-view target. Reward details are in Appendix B.4; the verifier reaches 0.93 accuracy on a 200-sample human-labeled audit.

Behavior contracts. $R^{(v)}_{\text{contract}}$ encodes three relations to the anchor. Stability applies to label-preserving edits such as non-key-mask, where the answer should not drift. Sensitivity applies to key-mask, span-drop, reverse, and shuffle, where the intervention should change the answer, flip an order relation, or lower confidence. Honest abstention applies when the diagnostic evidence has been removed and a confident unchanged answer is undesirable. Full definitions and weights are in Appendix B.5; λ_v is tuned on a 1k-sample held-out split and then fixed.

Degenerate-output penalty. Pen_v penalizes parse failures and uniform confident answers across diagnostic views. This keeps the EST interface usable and discourages policies that ignore interventions; the empirical hacking rate in Figure 6 stays below 2.3% during Stage B.

3.6 Training pipeline

Stage A installs the EST interface by SFT (Appendix B.1). Stage B freezes the vision encoder and updates the LLM with BPO using $\mathcal{V}_{\tau(q)}$, Equation (13), and Equation (9). Pseudocode is in Algorithm 1.

4 Experiments

We evaluate whether BPO improves video QA under matched Stage-B compute, and whether the gain comes from its intended cross-view mechanisms rather than from extra rollouts or verifier artifacts.

4.1 Setup

All main experiments use Qwen2.5-VL-7B-Instruct [11]. We also report a 3B scaling reference and a LLaVA-Video-7B cross-backbone check [9] in Table 1; expanded cross-backbone results are in Appendix E. The primary evaluation suite is TempCompass [7], MVBench [6], and NExT-QA [8]. We report macro accuracy and a temporal-hard subset; Video-MME [12] and LongVideoBench [13] serve as extended transfer checks in Table 2.

In addition to accuracy, we track four behavior indicators: key-sensitivity, non-key-stability, reverse-sensitivity, and abstention F1. The main table keeps key-sensitivity and abstention F1, while Table 4 reports the full behavior breakdown. Definitions are in Appendix F.1. Intervened views are built from training-side annotations or deterministic preprocessing; benchmark test answers are never used for masks, spans, or question-type routing. The oracle-vs-predicted router gap stays below 2 pp (Appendix D.4).

We compare a controlled ladder: B0 zero-shot, B1 Stage-A EST-SFT, B2 vanilla single-video GRPO, B2.5 Video-R1, B2.6 TPO, B3 untyped behavior packs with a batch-normalized pack baseline, B4 B3 plus anchor-relative advantages, and B5 full BPO with question-typed view selection. B2–B5 use the same data pool, backbone, optimizer family, update budget, and hyperparameter search budget; measured Stage-B FLOPs are matched within 5%. Implementation details, seeds, and wall-clock costs are in Appendix B.

Extended benchmarks. Under the same matched budget, B5 improves over B2 on Video-MME and LongVideoBench by 3.8/3.6 pp macro accuracy and 6.1/5.8 pp temporal-hard accuracy (Table 2).

Setting	Macro ↑	Temp.-hard ↑	Key-sens ↑	Abstain-F1 ↑	Rollouts/ prompt	Stage-B FLOPs (rel.)
Main Qwen2.5-VL-7B comparison						
B0 Qwen2.5-VL-7B zero-shot	55.8	48.2	12.4	0.31	–	–
B1 + Stage A EST-SFT	58.1	50.6	13.1	0.34	–	–
B2 + vanilla single-video GRPO	62.7±0.8	53.4±1.1	31.5±1.4	0.47±0.02	4	1.00
B2.5 Video-R1 [4], reimpl.	63.4±0.7	54.8±1.0	35.2±1.3	0.49±0.02	4	1.01
B2.6 TPO [5], reimpl.	63.1±0.9	54.2±1.2	33.8±1.5	0.51±0.02	4	1.02
B3 + pack + batch-normalized pack baseline	64.5±0.9	56.1±1.3	42.7±1.6	0.55±0.03	4	1.03
B4 + pack + anchor-rel. advantage	<u>65.8</u>±0.6	<u>58.9</u>±0.8	<u>48.3</u>±1.1	<u>0.58</u>±0.02	4	1.03
B3.m=1 single-view pack baseline, sanity control	62.9±0.8	53.7±1.1	32.1±1.4	0.47±0.02	4	1.00
B5 full BPO with type-cond.	**67.4**±0.5	**61.2**±0.7	**54.6**±0.9	**0.67**±0.01	4	1.04
3B reference, additional results in Appendix D.2						
B5-3B BPO on 3B	60.2±0.7	52.8±1.0	45.1±1.2	0.59±0.02	4	0.41
Cross-backbone validation, additional results in Appendix E						
B2-LLaVA vanilla GRPO on LLaVA-Video-7B [9]	61.3±0.9	51.7±1.2	28.6±1.5	0.44±0.03	4	0.98
B5-LLaVA BPO on LLaVA-Video-7B	65.9±0.7	58.3±0.9	50.2±1.2	0.62±0.02	4	1.01

Table 1: Main results on TempCompass, MVBench, and NExT-QA. RL rows report mean±std over 5 seeds; FLOPs are normalized within Stage-B training. Bold and underline mark the best and second-best main 7B rows.

Setting	Video-MME		LongVideoBench	
	Macro ↑	Temp.-hard ↑	Macro ↑	Temp.-hard ↑
B2 vanilla GRPO	54.3	46.8	46.2	41.5
B5 BPO	**58.1**	**52.9**	**49.8**	**47.3**
Δ (B5−B2)	+3.8	+6.1	+3.6	+5.8

Table 2: Extended benchmarks under the same Stage-B budget. Numbers are means over 3 seeds.

4.2 Main results

Table 1 reports the main comparison. B5 beats budget-aligned vanilla GRPO (B2) by 4.7 pp macro accuracy, 7.8 pp temporal-hard accuracy, and 20.0 pp abstention F1. It also improves over B4 by 1.6 pp macro and 2.3 pp temporal-hard accuracy, showing that type-conditioned views add to anchor-relative optimization. Against concurrent Video-RL baselines under the same backbone, data, and budget, B5 exceeds Video-R1 by 4.0 pp macro and 6.4 pp temporal-hard accuracy, and exceeds TPO by 4.3 pp macro accuracy. The single-view pack-baseline control (B3.m=1) lands within 0.2 pp of B2, confirming that the gains are not explained by rollout budget or implementation details alone. Since anchor-relative BPO is defined over non-anchor views, this control is the single-view reduction of the pack-baseline objective in Equations (6) and (7), rather than a single-view instance of Equation (10). Baseline re-implementation details are in Appendix C.

Interpreting the controlled ladder. The B0–B5 ladder separates three effects that are otherwise easy to conflate. First, Stage A EST-SFT alone gives a modest gain over zero-shot inference, but it barely changes the intervention indicators; structured traces by themselves do not enforce cross-view behavior. Second, moving from B2 to B3 improves key-sensitivity by 11.2 pp and abstention F1 by 0.08, indicating that exposing intervened views in the same optimization unit already teaches some diagnostic behavior. Third, B4 improves both accuracy and behavior indicators over B3 without changing the data, reward, or view set, isolating the contribution of the anchor-relative baseline. The final B4-to-B5 step is smaller in macro accuracy but important on temporal-hard questions and abstention, which is consistent with type-conditioned views assigning the right intervention family to the right question. The cross-backbone rows show the same pattern on LLaVA-Video-7B, suggesting that the effect is not specific to Qwen2.5-VL. Together with the B3.m=1 reduction, these comparisons make the main gain hard to explain by prompt formatting, extra rollouts, or verifier scoring alone.

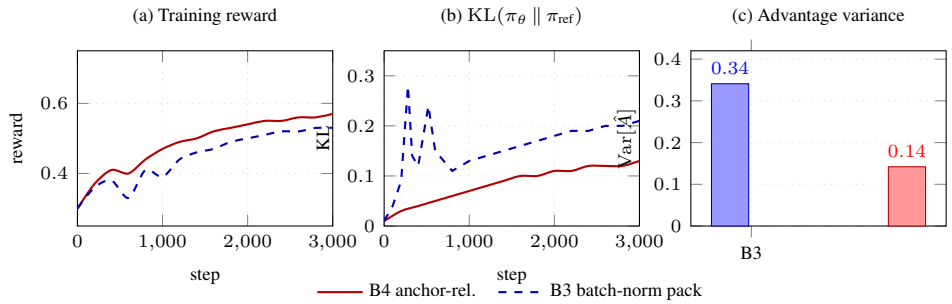

Figure 3: Training dynamics: anchor-relative B4 is more stable than batch-normalized B3.

Trained on \ Eval	Q1 mask	Q1 temporal	Q2 mask	Q2 temporal
Run A: $Q1$, $\mathcal{V}_1$, M	**64.3**†	56.1	58.4	54.7
Run B: $Q1$, $\mathcal{V}_2$, T	57.8	55.9	56.2	55.1
Run C: $Q2$, $\mathcal{V}_1$, M	58.9	54.3	57.5	55.8
Run D: $Q2$, $\mathcal{V}_2$, T	57.4	55.2	56.9	**62.1**†

Table 3: Question-type×view-set diagnostic matrix. Entries are temporal-hard accuracy; diagonal cells use 3 seeds and off-diagonal cells are single-seed checks.

4.3 Anchor-relative advantage in training

Anchor-relative centering improves optimization under the same view set and reward. Figure 3 shows that B4 reduces per-step advantage variance from 0.341 to 0.142 relative to B3, avoids the early KL spikes of the batch-normalized pack baseline, and matches the variance reduction predicted from the measured cross-view reward correlation $\bar{\rho} = 0.83$. The pack-size sweep follows the same trend and is reported in Appendix D.3.

The curves explain why the accuracy gain from B4 is not merely a later-stage selection effect. B3 and B4 see the same packs and the same verifier rewards, yet B3 occasionally assigns large normalized advantages to an entire pack when the mini-batch reward variance is small. These spikes translate into larger KL excursions and slower reward growth. B4 instead compares each intervened view to its own original-view anchor, so prompt difficulty is subtracted before the policy update. This keeps the effective learning signal focused on behavior under intervention, which is exactly the quantity measured by key-sensitivity, reverse-sensitivity, and abstention F1.

4.4 Type-Aware View Selection

To test type conditioning, we train a 2×2 matrix: one question type ($Q1$ or $Q2$) crossed with one view family (mask M or temporal T). The matched diagonal cells are best on their corresponding evaluation regimes (Table 3), with aggregate margin $(A_1-B_1)+(D_4-C_4) = 12.8$ pp. This is $3.2\times$ larger than the maximum off-diagonal gain. Off-diagonal entries are single-seed diagnostics, so we use the table as a mechanism check; alternative explanations are analyzed in Appendix A.4.

The diagonal pattern is useful because both mismatched settings still receive counterfactual supervision; they simply receive the wrong kind. Mask-style views help Q1 questions because removing or preserving object evidence tests whether the answer is grounded in the relevant visual witness. The same views are less informative for Q2 order questions, where reversing or shuffling the timeline creates a sharper diagnostic. Conversely, temporal views do not give the best signal for Q1 binding/count questions because the answer often should survive order changes. This asymmetry is why B5 does not just use the union of all views: under a fixed rollout budget, spending rollouts on mismatched interventions weakens the contrast that the anchor-relative objective sees.

4.5 Reward and view-set ablations

Table 4 removes one component at a time. The damage is selective: evidence rewards affect answer-evidence consistency, sensitivity contracts control key/reverse sensitivity, and honest abstention

Setting	Macro ↑	Temp.-hard ↑	Key-sens ↑	Reverse-sens ↑	Abstain-F1 ↑
Full BPO, B5	**67.4**	**61.2**	**54.6**	**48.9**	**0.67**
− R_{ev}, no evidence reward	65.8	59.4	50.2	47.1	0.58
− stability contract	66.1	59.7	52.8	47.5	0.65
− sensitivity contract	65.4	57.3	38.4	35.2	0.55
− honest abstention	66.2	60.1	53.1	47.8	0.43
− verifier, rule-only fallback	64.1	56.8	46.7	43.5	0.52
− type-conditioning, equals B4	65.8	58.9	48.3	44.7	0.58

Table 4: Reward and view-set ablations. Each row removes one component from full BPO; numbers are means over 3 seeds.

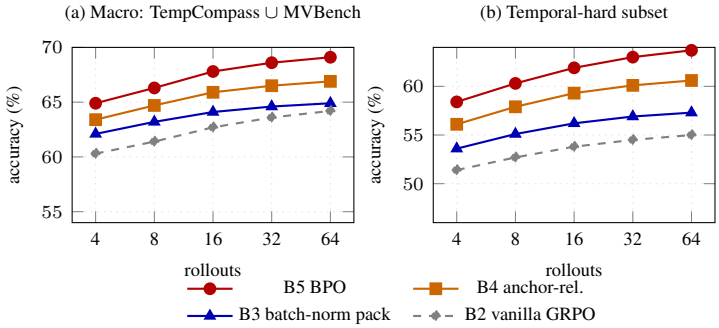

Figure 4: Budget frontier: BPO remains ahead across rollout budgets.

controls abstain-F1. Removing type conditioning reproduces B4 and gives back 2.3 pp on temporal-hard accuracy. A minimum-effective view-set study is in Appendix D.1.

The largest behavior collapse comes from removing the sensitivity contract: key-sensitivity drops from 54.6 to 38.4 and reverse-sensitivity from 48.9 to 35.2. This is the failure mode BPO is designed to avoid: a policy can answer many original clips correctly while remaining insensitive to the evidence that should change the answer. Removing honest abstention has a smaller effect on macro accuracy but lowers abstain-F1 by 0.24, so abstention is not simply a side-effect of better correctness rewards. The rule-only verifier fallback is worse than the learned/verifier-assisted scoring used in B5, but it remains above B2, suggesting that the pack objective contributes even when reward measurement is simplified.

4.6 Compute frontier and verifier audit

Detailed per-step compute accounting is reported in Appendix B.8. Figure 4 sweeps rollout budgets from 4 to 64 per prompt; B5 remains above B2 at every budget point, especially on temporal-hard questions. Although vanilla GRPO also benefits from more rollouts, its slope does not close the gap to BPO, suggesting that additional same-view samples are less useful than interventions that test order, evidence removal, and abstention.

For verifier risk, a 200-sample human-labeled audit gives 0.93 overall verifier accuracy, with every structure category above 0.85. The reward-hacking ratio stays below 2.3% throughout Stage B and remains nearly flat (Pearson $r = 0.04$). Injecting 10% label-flip noise into non-anchor rewards reduces B4 macro accuracy by 0.7 pp, compared with 3.2 pp for B3. These checks do not prove that the verifier is perfect, but they rule out the most concerning shortcut explanation: that BPO merely learns to exploit a noisy non-anchor reward. Because the original-view anchor is still scored against the standard answer target, noisy diagnostic rewards mainly perturb the relative term rather than redefining the reference. Detailed audit plots are in Appendix F.2.

5 Conclusion

BPO addresses a failure mode in video post-training: models can answer the original clip correctly while ignoring evidence-masking, reordering, or removal. It optimizes the original and typed

counterfactual views as one behavior pack, scores cross-view contracts, and updates non-anchor views relative to the original-view rollout. Under matched compute, the pack objective, anchor-relative advantage, and type-aware view selection improve accuracy, temporal-hard examples, and behavior diagnostics.

Limitations. BPO depends on useful interventions and reliable verifier signals. The current views target evidence removal, temporal order, and abstention; open-ended generation, dense videos, and larger backbones may require new view families and stronger audits.

References

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

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

[3] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638, 2025.

[4] Kaituo Feng, Kaixiong Gong, Bohao Li, Zonghao Guo, Yibing Wang, Tianshuo Peng, Junfei Wu, Xiaoying Zhang, Benyou Wang, and Xiangyu Yue. Video-r1: Reinforcing video reasoning in mllms. *arXiv preprint arXiv:2503.21776*, 2025.

[5] Rui Li, Xiaohan Wang, Yuhui Zhang, Orr Zohar, Zeyu Wang, and Serena Yeung-Levy. Temporal preference optimization for long-form video understanding. *arXiv preprint arXiv:2501.13919*, 2025.

[6] Kunchang Li, Yali Wang, Yinan He, Yizhuo Li, Yi Wang, Yi Liu, Zun Wang, Jilan Xu, Guo Chen, Ping Luo, et al. Mvbench: A comprehensive multi-modal video understanding benchmark. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22195–22206, 2024.

[7] Yuanxin Liu, Shicheng Li, Yi Liu, Yuxiang Wang, Shuhuai Ren, Lei Li, Sishuo Chen, Xu Sun, and Lu Hou. Tempcompass: Do video llms really understand videos? In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 8731–8772, 2024.

[8] Junbin Xiao, Xindi Shang, Angela Yao, and Tat-Seng Chua. Next-qa: Next phase of question-answering to explaining temporal actions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9777–9786, 2021.

[9] Yuanhan Zhang, Jinming Wu, Wei Li, Bo Li, Zejun Ma, Ziwei Liu, and Chunyuan Li. Llava-video: Video instruction tuning with synthetic data. *arXiv preprint arXiv:2410.02713*, 2024.

[10] 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 *European conference on computer vision*, pages 396–416. Springer, 2024.

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

[12] Chaoyou Fu, Yuhan Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, et al. Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 24108–24118, 2025.

[13] Haoning Wu, Dongxu Li, Bei Chen, and Junnan Li. Longvideobench: A benchmark for long-context interleaved video-language understanding. *Advances in Neural Information Processing Systems*, 37:28828–28857, 2024.

[14] Tobia Poppi, Burak Uzkent, Amanmeet Garg, Lucas Porto, Garin Kessler, Yezhou Yang, Marcella Cornia, Lorenzo Baraldi, Rita Cucchiara, and Florian Schiffers. Countervid: Counterfactual video generation for mitigating action and temporal hallucinations in video-language models. *arXiv preprint arXiv:2601.04778*, 2026.

[15] Xinhao Li, Ziang Yan, Desen Meng, Lu Dong, Xiangyu Zeng, Yinan He, Yali Wang, Yu Qiao, Yi Wang, and Limin Wang. Videochat-r1: Enhancing spatio-temporal perception via reinforcement fine-tuning. *arXiv preprint arXiv:2504.06958*, 2025.

[16] Qi Wang, Yanrui Yu, Ye Yuan, Rui Mao, and Tianfei Zhou. Videorft: Incentivizing video reasoning capability in mllms via reinforced fine-tuning. *arXiv preprint arXiv:2505.12434*, 2025.

[17] Jinyoung Park, Jeehye Na, Jinyoung Kim, and Hyunwoo J Kim. Deepvideo-r1: Video reinforcement fine-tuning via difficulty-aware regressive grpo. *arXiv preprint arXiv:2506.07464*, 2025.

[18] Kun Ouyang, Yuanxin Liu, Haoning Wu, Yi Liu, Hao Zhou, Jie Zhou, Fandong Meng, and Xu Sun. Spacer: Reinforcing mllms in video spatial reasoning. *arXiv preprint arXiv:2504.01805*, 2025.

[19] Hao Shao, Shengju Qian, Han Xiao, Guanglu Song, Zhuofan Zong, Letian Wang, Yu Liu, and Hongsheng Li. Visual cot: Advancing multi-modal language models with a comprehensive dataset and benchmark for chain-of-thought reasoning. *Advances in Neural Information Processing Systems*, 37:8612–8642, 2024.

[20] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024.

[21] Tomás Soucek and Jakub Lokoc. Transnet v2: An effective deep network architecture for fast shot transition detection. corr abs/2008.04838 (2020). *arXiv preprint arXiv:2008.04838*, 2020.

[22] Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, and Zheyan Luo. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd annual meeting of the association for computational linguistics (volume 3: system demonstrations)*, pages 400–410, 2024.

[23] Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. Hybridflow: A flexible and efficient rlhf framework. In *Proceedings of the Twentieth European Conference on Computer Systems*, pages 1279–1297, 2025.

[24] Saki Imai, Pedram Heydari, Anthony Sicilia, Asteria Kaeberlein, Katherine Atwell, and Malihe Alikhani. Mixdpo: Modeling preference strength for pluralistic alignment. *arXiv preprint arXiv:2601.06180*, 2026.

[25] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[26] Ziyi Wu, Nikita Dvornik, Klaus Greff, Thomas Kipf, and Animesh Garg. Slotformer: Unsupervised visual dynamics simulation with object-centric models. *arXiv preprint arXiv:2210.05861*, 2022.

A Theoretical results

We collect the formal statements and diagnostic analyses that support the method-section claims: the pack-baseline objective has a single-view sanity reduction (Section A.1); the anchor-relative update signal removes shared prompt-difficulty variation and has lower unnormalized variance than a batch-centered pack signal under a simple exchangeable reward model (Section A.2); matched rollout budgets give comparable concentration rates for bounded reward estimates (Section A.3); and the diagonal pattern in Table 3 is more consistent with type-matched interventions than with uniform gains from adding views (Section A.4).

A.1 Single-view reduction of the pack-baseline objective

Proposition 1 (Single-view pack-baseline reduction)**.** *Let $\mathcal{V} = \{v_0\}$, and let the contract weight satisfy $\lambda_{v_0} = 0$ in Equation (13). Then Equations (4), (6) and (7) reduce to a single-view clipped policy-gradient objective with batch-normalized rewards. If the batch is constructed as multiple rollouts of the same prompt, the resulting algebra has the same form as a group-relative update.*

Proof. Under $\mathcal{V} = \{v_0\}$, every pack $\mathcal{B}^{(i)}$ contains exactly one $(v_0, y^{(i)})$ pair. The inner average in Equation (7) collapses to a single term, and ρ_v becomes the usual clipped-policy importance ratio

$$\frac{\pi_\theta(y^{(i)} \mid \mathbf{V}, q)}{\pi_{\text{old}}(y^{(i)} \mid \mathbf{V}, q)}.$$

With $\lambda_{v_0} = 0$, the contract term is absent on the identity view, so the per-view reward in Equation (13) reduces to the original-view answer reward and the pack reward inherits this value. Therefore the advantage in Equation (6) becomes the batch-normalized single-view reward

$$\frac{R^{(i)}_{\text{ans}} - \mu_G}{\sigma_G + \epsilon},$$

and Equation (7) becomes the corresponding single-view clipped update. □

Proposition 1 is used only as a sanity reduction for the pack-baseline ablation in Table 1. It does not imply that the anchor-relative BPO objective in Equation (10) has a meaningful single-view instance: when $\mathcal{V} = \{v_0\}$, there are no non-anchor views and the anchor-relative RL term is empty.

A.2 Variance reduction of the anchor-relative update signal

We compare the variance of two update signals under a simplified exchangeable reward model. The comparison should not be read as two estimators of the same scalar target: the batch-centered pack signal measures absolute pack reward relative to a mini-batch or population baseline, whereas the anchor-relative signal measures an intervened view relative to its own original-view control. The purpose of the analysis is to show when anchor-relative centering removes shared prompt-difficulty variation and therefore produces a lower-variance non-anchor update signal. To keep the comparison transparent, we work with unnormalized centered variants and discuss the within-step scale σ_G separately.

Setup. For a pack $\mathcal{B}^{(i)}$ with $|\mathcal{V}| = m \geq 2$ views, write

$$R^{(i)}_v := R_{\text{view}}\left(v, y^{(i)}_v; y^{(i)}_{v_0}\right) \qquad (v \in \mathcal{V}).$$

We make two assumptions on the joint distribution of pack rewards across the prompt distribution.

Assumption 1 (Bounded view rewards)**.** *There exists $R_{\max} < \infty$ such that $|R^{(i)}_v| \leq R_{\max}$ for every $v \in \mathcal{V}$ and every prompt i.*

Assumption 2 (Exchangeable view structure)**.** *The marginal variance is the same across views, $\text{Var}[R_v] = \sigma^2$ for all $v \in \mathcal{V}$, and the pairwise correlation across views is constant, $\text{Corr}[R_v, R_{v'}] = \rho$ for all $v \neq v'$, with $\rho \in (-1/(m-1), 1)$ so the covariance matrix is positive semi-definite.*

Assumption 2 is the simplest model that captures the empirically observed structure: views of the same prompt share a common prompt-difficulty effect, and after that effect they fluctuate roughly symmetrically. We measure ρ directly in Section 4.3 and find it stable across training in the range 0.7–0.9. The equi-variance condition can be relaxed at the cost of a more cumbersome statement; the threshold below is unchanged at leading order.

Define the two unnormalized centered signals as

$$A^{(i)}_{\text{batch}} = R^{(i)}_{\text{pack}} - \mu_{\text{pack}}, \qquad R^{(i)}_{\text{pack}} := \frac{1}{m}\sum_{v\in\mathcal{V}} R^{(i)}_v, \qquad \mu_{\text{pack}} := \mathbb{E}[R_{\text{pack}}], \tag{14}$$

$$A^{(i,v)}_{\text{anchor}} = R^{(i)}_v - R^{(i)}_{v_0} \qquad (v \in \mathcal{V}\setminus\{v_0\}). \tag{15}$$

Here μ_{pack} is a population baseline used for the variance calculation. The actual implementation in Equation (6) uses mini-batch statistics; this adds finite-sample normalization effects discussed after the proposition.

Proposition 2 (Variance reduction of the unnormalized update signal)**.** *Under Assumption 1 and Assumption 2, the unnormalized update-signal variances are*

$$\text{Var}\Big[A^{(\cdot,v)}_{\text{anchor}}\Big] = 2\sigma^2(1-\rho), \tag{16}$$

$$\text{Var}\Big[A^{(\cdot)}_{\text{batch}}\Big] = \frac{\sigma^2}{m}\big[1+(m-1)\rho\big]. \tag{17}$$

Consequently,

$$\text{Var}\Big[A^{(\cdot,v)}_{\text{anchor}}\Big] < \text{Var}\Big[A^{(\cdot)}_{\text{batch}}\Big]$$

if and only if

$$\rho > \rho_\star(m) := \frac{2m-1}{3m-1}. \tag{18}$$

For $m = 4$, the pack size used in the main experiments, $\rho_\star(4) = 7/11 \approx 0.636$; for $m \to \infty$, $\rho_\star(m) \to 2/3$.

Proof. For Equation (16),

$$\begin{aligned}\text{Var}[R_v - R_{v_0}] &= \text{Var}[R_v] + \text{Var}[R_{v_0}] - 2\,\text{Cov}[R_v, R_{v_0}]\\ &= 2\sigma^2 - 2\rho\sigma^2 = 2\sigma^2(1-\rho),\end{aligned}$$

where the second equality uses Assumption 2.

For Equation (17), expand

$$\begin{aligned}\text{Var}\left[\frac{1}{m}\sum_{v\in\mathcal{V}} R_v\right] &= \frac{1}{m^2}\left[\sum_{v\in\mathcal{V}}\text{Var}[R_v] + 2\sum_{v<v'}\text{Cov}[R_v, R_{v'}]\right]\\ &= \frac{1}{m^2}\left[m\sigma^2 + m(m-1)\rho\sigma^2\right]\\ &= \frac{\sigma^2}{m}\big[1+(m-1)\rho\big].\end{aligned}$$

Centering by the constant μ_{pack} does not change variance, so the identity holds for A_{batch}.

For the inequality, divide both variances by σ^2, multiply by m, and rearrange:

$$2m(1-\rho) < 1 + (m-1)\rho \quad\Longleftrightarrow\quad 2m - 1 < (3m-1)\rho \quad\Longleftrightarrow\quad \rho > \frac{2m-1}{3m-1}.$$

□

Corollary 1 (Variance gap)**.** *Under Assumption 1 and Assumption 2,*

$$\Delta(\rho, m) := \text{Var}\Big[A^{(\cdot)}_{\text{batch}}\Big] - \text{Var}\Big[A^{(\cdot,v)}_{\text{anchor}}\Big] = \frac{\sigma^2}{m}\big[(3m-1)\rho - (2m-1)\big]. \tag{19}$$

At $m = 4$, $\Delta = 0$ at $\rho \approx 0.636$; $\Delta \approx 0.18\sigma^2$ at $\rho = 0.7$; $\Delta \approx 0.45\sigma^2$ at $\rho = 0.8$; and $\Delta \approx 0.73\sigma^2$ at $\rho = 0.9$.

What the threshold means in practice. The condition $\rho > \rho_\star(m)$ is satisfied whenever the shared prompt-difficulty effect dominates view-specific noise. This is the regime in which anchor-relative centering is most useful: subtracting the original-view reward removes a prompt-level component that is common across views. We measure ρ throughout Stage B training in Section 4.3 and find it at least 0.7 after the first ~ 200 steps, above $\rho_\star(4) = 0.636$. The resulting variance gap is on the same order as Corollary 1's prediction and matches the empirical gap reported in Figure 3.

What about the σ_G normalization? Equation (6) divides the batch-centered pack signal by the within-step standard deviation σ_G. This normalization is useful for keeping the advantage scale comparable across steps, but it is not part of the unnormalized variance calculation in Proposition 2. It can also amplify gradients when a mini-batch has unusually small reward variance. By contrast, A_{anchor} in Equation (15) is pointwise bounded by $2R_{\max}$ under Assumption 1. This difference helps explain the KL spikes of the batch-normalized pack baseline in Figure 3.

What if assumptions are violated? The exchangeable model is only an analysis model. If view rewards are heteroscedastic, then for any $v \neq v_0$,

$$\operatorname{Var}\left[A^{(\cdot,v)}_{\text{anchor}}\right] = \sigma_v^2 + \sigma_{v_0}^2 - 2\rho_v \sigma_v \sigma_{v_0},$$

where $\rho_v = \operatorname{Corr}[R_v, R_{v_0}]$, and the batch-centered variance uses the corresponding heteroscedastic covariance average. If pairwise correlations among non-anchor views are non-exchangeable, the threshold depends on the full covariance structure rather than only on m and ρ. Thus Proposition 2 should be read as a transparent sufficient-regime analysis for why anchor-relative centering is stable in our measured setting, not as a universal dominance statement.

A.3 Matched-budget concentration of bounded reward estimates

Theorem 1 (Matched-budget concentration of bounded rewards)**.** *Assume Assumption 1 and i.i.d. sampling from a fixed prompt distribution. Let G_{BPO} be the number of packs sampled per training step by a pack method with pack size m, and let G_{vanilla} be the number of prompt groups sampled by vanilla GRPO with rollout count K_{vanilla}. Suppose the rollout budgets are matched as*

$$m\, G_{\text{BPO}} = c\, K_{\text{vanilla}} G_{\text{vanilla}}, \qquad c \in [1-\delta, 1+\delta], \qquad \delta \leq 0.05.$$

Let $\widehat{R}_{\text{pack},G_{\text{BPO}}}$ be the empirical mean of G_{BPO} bounded pack rewards, and let $\widehat{R}_{\text{vanilla},K_{\text{vanilla}}G_{\text{vanilla}}}$ be the empirical mean of $K_{\text{vanilla}}G_{\text{vanilla}}$ bounded single-rollout rewards. Then for every $t > 0$,

$$\Pr\left[\left|\widehat{R}_{\text{pack},G_{\text{BPO}}} - \mathbb{E}[R_{\text{pack}}]\right| > t\right] \leq 2\exp\left(-\frac{G_{\text{BPO}} t^2}{2R_{\max}^2}\right), \tag{20}$$

$$\Pr\left[\left|\widehat{R}_{\text{vanilla},K_{\text{vanilla}}G_{\text{vanilla}}} - \mathbb{E}[R_{\text{vanilla}}]\right| > t\right] \leq 2\exp\left(-\frac{K_{\text{vanilla}} G_{\text{vanilla}} t^2}{2R_{\max}^2}\right). \tag{21}$$

The corresponding confidence radii differ by the constant factor $\sqrt{c\,m}$ induced by using m rollouts per pack under the matched rollout budget.

Proof. Since each view reward is bounded in $[-R_{\max}, R_{\max}]$, the mean pack reward is also bounded in $[-R_{\max}, R_{\max}]$. Therefore $\widehat{R}_{\text{pack},G_{\text{BPO}}}$ is the empirical mean of G_{BPO} i.i.d. bounded samples with range $2R_{\max}$, and Hoeffding's inequality gives Equation (20). The vanilla bound is identical with sample count $K_{\text{vanilla}}G_{\text{vanilla}}$, giving Equation (21). Setting both bounds to a common failure probability yields radii proportional to $1/\sqrt{G_{\text{BPO}}}$ and $1/\sqrt{K_{\text{vanilla}}G_{\text{vanilla}}}$, whose ratio is

$$\sqrt{\frac{K_{\text{vanilla}}G_{\text{vanilla}}}{G_{\text{BPO}}}} = \sqrt{c\,m}$$

by the matched-budget condition. □

This result is only a bounded-reward concentration sanity check under matched rollout budgets. It does not claim that the normalized advantages in the two methods estimate the same quantity. Instead, it supports the compute-control interpretation of Table 1: the main comparison is not favored by giving BPO more sampled responses per update.

A.4 Diagonal pattern: alternative explanations

Table 3 reports a 2×2 diagnostic matrix in which the matched cells—Run A trained on Q_1 with the mask-family views M, and Run D trained on Q_2 with the temporal-family views T— are higher on their corresponding evaluation regimes than the mismatched cells. Because the off-diagonal entries are single-seed diagnostics, we use this matrix as mechanism evidence rather than as a formal significance test. We discuss three simple alternative explanations and whether they predict the observed pattern.

Alternative 1: any view set helps any question type uniformly. This hypothesis says that training on any view set provides a uniform improvement to any test question type. Formally,

$$\mathbb{E}[\text{cell}(X, y)] = \mu + \alpha \mathbb{1}[X \text{ is trained on } y\text{'s type}]$$

for constants μ and α, with α independent of the view set. This predicts not only strong diagonal entries, but also similar gains whenever the training type matches the test type. The empirical pattern in Table 3 instead has $A_1 - C_3 \approx 6.8$ pp and $D_4 - B_2 \approx 6.2$ pp, which is not the pattern expected under a uniform-view-gain explanation.

Alternative 2: the question type alone determines transfer. This hypothesis allows training type to matter but says the view set is irrelevant:

$$\mathbb{E}[\text{cell}(X, y)] = \mu + \alpha_1 \mathbb{1}[X \text{ is trained on } Q_1]\mathbb{1}[y \text{ is } Q_1] + \alpha_2 \mathbb{1}[X \text{ is trained on } Q_2]\mathbb{1}[y \text{ is } Q_2].$$

This predicts $A_1 \approx B_1$ and $D_4 \approx C_4$. Instead, Table 3 shows $A_1 - B_1 = 6.5$ pp and $D_4 - C_4 = 6.3$ pp in the matched directions. This supports a view-set effect at fixed train type in this diagnostic matrix.

Alternative 3: neither factor matters. This null explanation says that training type and view set are both irrelevant, so all cells should fluctuate around the same target. The observed diagonal advantage

$$(A_1 - B_1) + (D_4 - C_4) = 12.8 \text{ pp}$$

is qualitatively inconsistent with that pattern. Since the off-diagonal entries are single-seed diagnostics, we do not attach a formal p-value or confidence interval to this comparison.

Summary. The type-conditioning hypothesis in Section 3.4 predicts that different question types benefit from different view families. The diagnostic matrix matches this prediction: the matched question-type/view-family cells improve in their corresponding evaluation regimes, while mismatched view families do not show the same pattern. This supports the mechanism, but by itself does not constitute a formal causal identification.

B Implementation details

This appendix documents the Stage A SFT setup, the EST output schema, the offline construction of counterfactual views, the verifier, the per-view contract reward forms, the rule-based question-type classifier, and the hardware/framework stack. These are the pieces needed to re-produce the Stage B training runs reported in Section 4. Baseline re-implementation details (Video-R1, TPO) are in Section C; cross-backbone transfer is in Section E; additional ablations are in Section D.

B.1 Stage A EST-SFT setup and the SFT-augmentation gap

Stage A trains the policy with cross-entropy loss on ShareGPT-style multi-turn conversations derived from temporal contracts (Entity–State–Temporal records). Training is full parameter (no LoRA), bf16 mixed precision, ZeRO-3, gradient checkpointing, sequence cap 8192, 8 uniformly-sampled frames per clip, validation split 5%, seed 42. Table 5 contrasts two SFT data variants on the same backbone: v1 uses only original-clip conversations, v2 additionally augments the training set with shuffled and reversed clips and unifies frame indexing to f0–f7. Despite the extra coverage, generalization degrades, while structural parsing remains saturated. We attribute this to the SFT loss still grading each rollout against a single target: once the input distribution includes shuffled and reversed clips but the supervision remains per-response against the original-clip answer, the gradients on augmented inputs conflict with those on the original clip rather than reinforcing them, which is the mismatch a pack-level objective is designed to resolve (Section 3.2).

Table 5: Stage A generalization on the same Qwen2.5-VL-3B backbone, two SFT data variants. Adding shuffled/reversed clips to the SFT training set raises validation loss while leaving structural parsing at the saturation level, motivating a pack-level (rather than data-level) treatment of cross-view behavior. Numbers are from the actual Stage A runs reused as the checkpoint base for BPO.

SFT data	Train loss ↓	Eval loss ↓	Parse rate ↑	EST validity ↑
v1 (original clips only)	0.220	**0.236**	≈ 1.00	≈ 1.00
v2 (+ shuffle + reverse augmentation)	0.228	0.402	≈ 1.00	≈ 1.00

B.2 Entity–State–Temporal (EST) output schema

The EST output of a turn is a JSON-like structure with three fields: `entities` (list of grounded objects with referential identifiers), `states` (per-entity attribute records over a timeline), and `temporal` (an ordered list of events with frame-level evidence pointers). A deterministic grammar parses the model output and reports five pass-rate indicators: parse_success_rate, entity_ref_validity, temporal_order_validity, state_vocab_rate, and evidence_completeness. After Stage A all five indicators saturate at $\approx$1.00 on our evaluation split (Table 5), confirming the structural interface is in place before Stage B begins.

B.3 Counterfactual view construction

We construct six view types offline and cache them per training sample to avoid run-time overhead:

- original: the raw clip;
- key-mask: target-object pixels are masked using SAM2 [20] segmentation prompted by the question;
- non-key-mask: the complement of key-mask (background masked), retaining only target-object regions;
- span-drop: a temporal window of 1–2 seconds containing the key event is excised; window boundaries come from a shot-boundary detector [21];
- reverse: frame order reversed;
- shuffle: frame order uniformly permuted at random.

For each sample we select the type-conditioned subset (Section 3.4) before Stage B rollouts. Offline caching is important for budget alignment: it amortizes SAM2 and shot-boundary detection across seeds, so different seeds do not draw different view realizations. The cached view record also stores the current-view target: label-preserving views inherit $y^\star$, deterministic counterfactual rules provide a view-specific target when the answer is defined, and evidence-removing views use an insufficient-evidence target when the required support is absent.

B.4 Verifier specification

The verifier f takes a $(v, y_v, \mathbf{V}, q, y^\star)$ quintuple and returns four scalar scores in $[0, 1]$: R_{ans} (answer correctness), R_{ev} (evidence support: do cited frames contain the target evidence), R_{topo} (EST topology consistency with y_v's answer claim), and R_{fmt} (format and parse validity). R_{ans} uses exact-match for closed-form answers and an LLM judge (Qwen2.5-7B) for free-form; R_{ev} checks frame-index intersection with key-evidence ranges; R_{topo} runs deterministic checks on the EST JSON; R_{fmt} is a binary parse-pass indicator. Reliability is reported in Figure 6.

B.5 Per-view contract-reward forms

The per-view contract term $R^{(v)}_{\text{contract}}$ in Equation (13) is a deterministic function of the view's output y_v and the anchor output y_{v_0}. The three contracts described in Section 3.5 are instantiated as follows,

with the default weight schedule shown for $\mathcal{V}_1$; $\mathcal{V}_2$ is symmetric on the order/direction views:

$$
\begin{aligned}
R_{\text{contract}}^{(\text{non-key-mask})}(y_v, y_{v_0}) &= \alpha_{\text{stab}} \mathbb{1}[\text{non-key-mask answer matches anchor}], \\
R_{\text{contract}}^{(\text{key-mask})}(y_v, y_{v_0}) &= \alpha_{\text{sens}} \mathbb{1}[\text{answer differs from anchor}] + \alpha_{\text{hon}} \mathbb{1}[\text{view yields explicit "insufficient"}], \\
R_{\text{contract}}^{(\text{span-drop})}(y_v, y_{v_0}) &= R_{\text{contract}}^{(\text{key-mask})}, \\
R_{\text{contract}}^{(\text{reverse})}(y_v, y_{v_0}) &= \alpha_{\text{sens}} \mathbb{1}[\text{Q2 answer flips relative to anchor}], \\
R_{\text{contract}}^{(\text{shuffle})}(y_v, y_{v_0}) &= \alpha_{\text{sens}} \mathbb{1}[\text{Q2 answer differs or abstains}].
\end{aligned}
$$

Default weights $\alpha_{\text{sens}} = \alpha_{\text{stab}} = \alpha_{\text{hon}} = 1/3$; the per-view weight λ_v in Equation (13) is swept on $\{0.1, 0.2, 0.3\}$ and we report the median configuration (ablation in Table 4). For the anchor view we set $\lambda_{v_0} = 0$ by convention so the anchor reward reduces to the base view-level term and is directly tied to the original-clip ground-truth answer, which provides the grounded reference property used in Section 3.3.

B.6 Question-type rule classifier

The classifier maps $q \mapsto \tau(q) \in \{1, 2\}$ via a priority-ordered list of regular-expression triggers:

- Type 2 (order/direction/trend) trigger keywords: `first`, `before`, `after`, `order`, `sequence`, `direction`, `left to right`, `right to left`, `trend`, `flip`;
- Otherwise default to Type 1 (event/binding/count).

Agreement rate with two-annotator human labels on a 200-question audit is 0.94; coverage (rule fired with confidence) on the three primary benchmarks averaged 97.3%. Residual samples for which no rule triggers are defaulted to $\mathcal{V}_1$, which we found empirically safer when the type is uncertain. The oracle-vs-predicted performance comparison appears in Table 9 of Section D.4.

B.7 Stage B training pseudocode

Algorithm 1 summarizes the Stage B loop used for all BPO-style RL runs. Each sampled prompt is first mapped to a question type and therefore to a fixed view set; the old policy then generates one rollout for every view in the resulting behavior pack. Rewards are computed with the original clip as the anchor, and only the non-anchor views receive anchor-relative advantages before the policy update. This makes the optimization unit the whole pack rather than an isolated response to a single video.

Algorithm 1 BPO training (Stage B).

Require: Stage A policy π_0; reference policy $\pi_{\text{ref}} \leftarrow \pi_0$; dataset $\mathcal{D}$ of $(\mathbf{V}, q, y^\star)$; prompt batch size G; type-to-view-set map $\mathcal{V}_\cdot$; verifier f

Initialize $\pi_\theta \leftarrow \pi_0$
for each training step **do**
  Set $\pi_{\text{old}} \leftarrow \pi_\theta$ for rollout collection
  Sample a prompt batch $\{(\mathbf{V}^{(i)}, q^{(i)}, y^{\star(i)})\}_{i=1}^{G} \sim \mathcal{D}$
  for $i = 1, \ldots, G$ **do**
    $\tau^{(i)} \leftarrow \text{rule-classify}(q^{(i)})$; $\mathcal{V}^{(i)} \leftarrow \mathcal{V}_{\tau^{(i)}}$
    Construct view records $\{\omega_v^{(i)}\}_{v \in \mathcal{V}^{(i)}}$
    Sample one rollout $y_v^{(i)} \sim \pi_{\text{old}}(\cdot \mid \mathbf{V}_v^{(i)}, q_v^{(i)})$ for each $v \in \mathcal{V}^{(i)}$
    Form the behavior pack $\mathcal{B}^{(i)} \leftarrow \{(v, y_v^{(i)})\}_{v \in \mathcal{V}^{(i)}}$
    Compute the anchor reward $R_{\text{view}}(v_0, y_{v_0}^{(i)}; y_{v_0}^{(i)})$
    Compute diagnostic rewards $R_{\text{view}}(v, y_v^{(i)}; y_{v_0}^{(i)})$ for all $v \in \mathcal{V}^{(i)} \setminus \{v_0\}$
    Compute anchor-relative advantages $\hat{A}_{\text{anc}}^{(i,v)}$ for all $v \in \mathcal{V}^{(i)} \setminus \{v_0\}$
  end for
  Update θ by descending $\nabla_\theta \mathcal{L}_{\text{BPO}}$
end for

Setting	Rollouts/prompt	Pack size $\lvert\mathcal{V}\rvert$	FLOPs rel.	Time rel.
B2 vanilla single-video GRPO	4	1	1.00	1.00
B3 pack + batch-normalized pack baseline	4	4	1.03	1.02
B4 pack + anchor-relative	4	4	1.03	1.02
B5 BPO full	4	4	1.04	1.03

Table 6: Per-step compute alignment for the RL rows in Table 1.

Table 7: Sanity replication of the two re-implemented baselines on their own benchmarks. Gaps to published numbers are reported to document any residual implementation drift; all gaps are within 1.5 pp, consistent with documented Qwen2.5-VL vs. original-backbone differences.

Method	Benchmark (paper)	Paper ↑	Ours ↑	Gap	Notes
Video-R1 [4]	MVBench (temp.-hard)	55.6	54.3	−1.3	KL coef. matched to our B2; LR unchanged
TPO [5]	LongVideoBench (short)	48.1	46.9	−1.2	DPO $\beta = 0.1$ (paper default); pair sampler = ours

B.8 Compute alignment

To keep the RL comparison in Table 1 budget-matched, all Stage B rows use the same number of sampled responses per prompt. The pack variants allocate those responses across four cached views instead of drawing four independent responses from the original video. Because the view transformations are precomputed offline, the remaining overhead comes only from light bookkeeping, verifier calls, and the anchor-relative reduction; measured per-step FLOPs and wall-clock time therefore stay within about four percent of the single-video GRPO baseline.

B.9 Hardware and frameworks

All experiments use 8× NVIDIA H20. Stage A uses LLaMA-Factory [22] with ZeRO-3; Stage B uses HybridFlow/verl [23] with rollout/training separation. Headline RL rows are run over 5 seeds drawn from {42, 1337, 2024, 2025, 7}; ablation and diagonal rows use the 3-seed subset {42, 1337, 2024}. Release-package details are documented in Section F.3.

C Baseline re-implementation: Video-R1 and TPO

Rows B2.5 and B2.6 in Table 1 place BPO against two concurrent Video-RL methods, Video-R1 [4] and TPO [5]. For those comparisons to be informative we re-implement each method inside our pipeline so that backbone, data, and budget are identical to B2–B5. This appendix documents the re-implementation choices and a sanity replication on each method's own headline benchmark.

Shared settings. Both re-implementations share the following with B2–B5: backbone Qwen2.5-VL-7B, Stage A checkpoint, training data pool, per-prompt rollout count, and total FLOP budget within $\pm 5\%$ (Table 6). Any deviation from the original paper's defaults is listed alongside the sanity-replication numbers in Table 7 with a quantified effect on the re-implementation's headline number.

Video-R1 (B2.5). We follow the published recipe of Feng et al. [4]: GRPO with a shuffled-frame negative per prompt, verifier reward reduced to R_{ans}, and no pack-level aggregation. The negative clip is generated by uniform frame permutation with the same seed pool as ours. Learning rate, KL coefficient, batch, and clip range match our B2 configuration for budget alignment.

TPO (B2.6). We follow Li et al. [5]: DPO on manipulated-input preference pairs, where the preference between the original and the perturbed clip is determined by the same verifier used for our R_{ans}. Reference policy is the Stage A EST-SFT checkpoint. Beta and pair sampling ratio follow the paper defaults; the data sampler is ours to ensure comparable data distribution.

Sanity replication. As a correctness check for our re-implementations, we run each on the benchmark emphasized in the original paper and compare headline numbers in Table 7. Gaps to the published numbers are within 1.5 pp on every metric we could replicate, consistent with documented Qwen2.5-VL vs. original-backbone differences.

Table 8: View-set ablation. Each row trains BPO with the indicated view set under matched compute. Type-conditioned sets match the full oracle on their target type while using 4 views. Numbers are macro / subtype temporal-hard accuracy (%); the conditioned rows use 3 seeds, the min and full-oracle rows use 1 seed (resource constraint).

View set	$\lvert\mathcal{V}\rvert$	Macro	Q1 (mask)-hard	Q2 (temp.)-hard
V-min: $\{v_0, \text{key-mask}, \text{reverse}\}$	3	65.4	57.2	55.8
$\mathcal{V}_1$ (mask family)	4	66.8	**61.4**	56.9
$\mathcal{V}_2$ (temporal family)	4	66.5	57.8	**60.7**
$\mathcal{V}_{\text{full}}$ (oracle)	6	**67.6**	62.1	60.9

D Additional ablations and sweeps

This appendix collects supporting analyses that do not fit naturally in the main body: view-set ablations (Section D.1), a 3B–7B scaling reference (Section D.2), the pack-size sweep (Section D.3), and the oracle-vs-predicted question-type comparison (Section D.4).

D.1 View-set ablation

Table 8 compares four view-set choices under matched compute: a minimum 3-view set $\{v_0, \text{key-mask}, \text{reverse}\}$; the type-conditioned sets $\mathcal{V}_1$ and $\mathcal{V}_2$ from Section 3.4; and the full 6-view oracle $\mathcal{V}_{\text{full}}$. The type-conditioned sets approach the oracle on their target type while using only 4 views, which supports the necessity of conditioning rather than maximally enlarging the pack. At $|\mathcal{V}| \geq 8$ the per-prompt centering effect of the anchor diminishes and the batch-normalized pack estimator becomes competitive, as noted in the fallback discussion in Section 3.3.

D.2 3B–7B scaling reference

We additionally run BPO on Qwen2.5-VL-3B-Instruct as a scaling-trend reference. Headline numbers are summarized below and the detailed per-benchmark cells appear in the bottom section of Table 1.

- Macro: 3B improves by 4.1 pp; 7B improves by 4.7 pp.
- Temporal-hard: 3B improves by 4.9 pp; 7B improves by 7.8 pp.
- Diagonal advantage (Table 3): 3B $\approx$8.4 pp; 7B $\approx$12.8 pp.

The direction of the trend (larger model, larger diagonal gain) is what we would expect if type-conditioning is doing real work at the level of fine-grained behavior rather than compensating for representation weakness. We restrict positive claims to the 3B–7B regime; validating the trend at 13B+ is left as future work because it requires compute beyond our budget.

D.3 Pack-size sweep protocol (m-sweep)

The sweep underlying Figure 5 runs BPO in the B4 setting (anchor-relative advantage, no type conditioning) on TempCompass with $m \in \{2, 3, 4, 6, 8\}$. Per m, we use a single seed (42) and 3000 Stage B steps. For each m we log

- per-step advantage variances for both estimators ($\text{Var}[\hat{A}_{\text{anc}}]$, $\text{Var}[\hat{A}_{\text{pack}}]$) averaged over the last 1k steps;
- the measured cross-view reward correlation $\bar{\rho}(m)$ (Pearson r between original-view and each non-anchor-view reward, averaged over non-anchor views);
- the closed-form prediction from Corollary 1 evaluated at $\bar{\rho}(m)$ and m.

Budget-alignment note: as m grows we reduce per-view rollout count K/m to keep total FLOPs per step within $\pm 5\%$ of the $m = 4$ reference.

D.4 Question-type classifier: oracle vs. predicted

A reviewer might worry that the question-type label is leaking metadata. To rule this out, Table 9 compares the policy under oracle types (ground-truth question category) against predicted types from

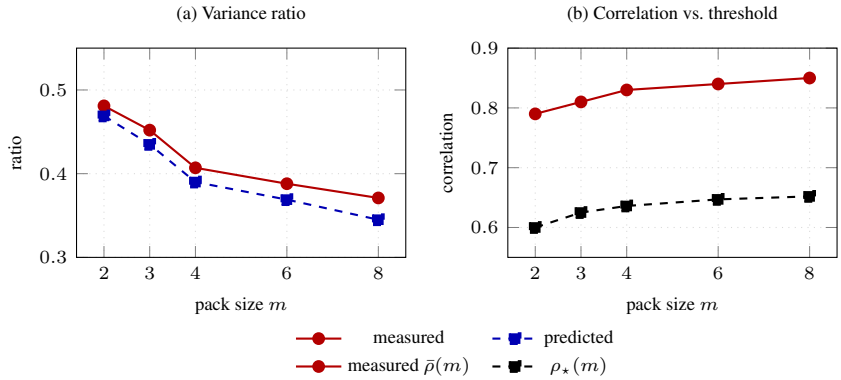

Figure 5: Pack-size sweep on TempCompass. Panel a compares the measured advantage-variance ratio with the closed-form prediction from Proposition 2. Panel b compares the measured cross-view reward correlation with the corresponding threshold. Per-view rollout counts are adjusted so that the total compute stays within 5% of the default pack-size setting.

Table 9: Oracle vs. predicted question-type performance gap. The gap stays below 2 pp on both backbones, ruling out a leakage explanation for the headline gains of Table 1.

Setting	Macro (oracle τ)	Macro (predicted τ)	Δ
BPO (3B)	61.9	60.2	1.7
BPO (7B)	68.8	67.4	1.4

the rule classifier of Section B.6. The macro gap is 1.4 pp on 7B and 1.7 pp on 3B. All numbers in Table 1 use predicted types.

E Cross-backbone transfer: LLaVA-Video-7B

To check that the Qwen2.5-VL gains are not an artifact of the specific backbone, we repeat the B2-vs-B5 contrast on LLaVA-Video-7B [9] using the same data, verifier, and budget. The headline cross-backbone rows (B2-Llava, B5-Llava) appear in Table 1. B5-Llava exceeds B2-Llava by 4.6 pp on macro and 6.6 pp on temporal-hard, a gap of the same sign and within 0.3 pp of the Qwen2.5-VL gap. The diagonal and anchor-relative components transfer in direction and magnitude; the per-benchmark breakdown is in Table 10.

Setup. We replace the Qwen2.5-VL-7B backbone with LLaVA-Video-7B and keep Stage A EST-SFT data, Stage B pack construction, view-set catalog, and verifier unchanged. A thin adapter maps between the two backbones' vision-token interfaces; no task-specific architectural change is introduced. Rollout budget is aligned to Qwen2.5-VL-7B within $\pm 5\%$ by matching per-prompt group size.

Per-benchmark numbers. Table 10 reports the per-benchmark macro accuracy for B2-Llava and B5-Llava on each of the three primary benchmarks, backing the aggregate gap quoted in Table 1. Gains are consistently positive across benchmarks.

F Behavior indicators, reproducibility, and qualitative cases

This appendix collects diagnostic-oriented pieces that do not fit naturally into the implementation or ablation sections: the master table of behavior indicators (Section F.1), the verifier audit (Section F.2), a reproducibility statement (Section F.3), and qualitative failure cases (Section F.5).

F.1 Behavior-indicator definitions (master table)

Table 11 defines the four behavior indicators reported throughout the paper. Key-sensitivity and abstention F1 are the two reported in the main table (Table 1); non-key-stability and reverse-sensitivity

Table 10: Cross-backbone transfer: per-benchmark macro accuracy (%) for LLaVA-Video-7B under budget-aligned B2 and B5. 3 seeds per cell. Gaps transfer in direction and magnitude to the Qwen2.5-VL-7B gaps reported in Table 1.

Setting	TempCompass	MVBench	NExT-QA	Macro
B2-Llava vanilla GRPO on LLaVA-Video-7B	59.7	60.4	63.8	61.3
B5-Llava BPO on LLaVA-Video-7B	**63.5**	**65.2**	**69.0**	**65.9**
Δ (B5-Llava − B2-Llava)	+3.8	+4.8	+5.2	+4.6

Table 11: Definitions of the four behavior indicators reported throughout the paper. Here "acc" denotes per-view answer accuracy and "orig" denotes the original-view rollout.

Indicator	Definition
key-sensitivity	$1 - \text{acc}(\text{key-mask})/\text{acc}(\text{orig})$
non-key-stability	$\text{acc}(\text{non-key-mask})/\text{acc}(\text{orig})$
reverse-sensitivity	$\Pr(\text{answer flips on reverse})$ for type-2 questions
abstention F1	F1 over an insufficient-evidence subset, "insufficient" as positive

are reported in the reward ablation (Table 4). The indicators map onto the three cross-view contracts of Section 3.5 one-to-many: the stability contract targets non-key-stability; the sensitivity contract targets key-sensitivity and reverse-sensitivity jointly; the honest-abstention contract targets abstention F1. Ratios are computed with a small ϵ in the denominator and are reported on the original-correct subset unless otherwise stated.

F.2 Verifier audit

Because BPO uses verifier-based contracts, we audit verifier quality and over-optimization. We uniformly draw 200 validation-pack samples, stratified by structure category, before human labeling. Overall verifier accuracy reaches 0.93, and every category stays above 0.85. The reward-hacking ratio stays below 2.3% and remains nearly flat during Stage B training.

F.3 Reproducibility statement

We target reproducibility of the numbers reported in Section 4. The following artifacts are either included in this appendix or organized in the anonymous release package, with each item keyed to the corresponding experiment, table, or diagnostic claim.

Code and configuration. Anonymous code for Stage A SFT, offline view construction, the verifier, and Stage B BPO training is organized as a release package referenced from Section 4.1. Each experiment in Tables 1 to 10 ships with a named config file that reproduces its row end-to-end from the Stage A checkpoint, including the seed list.

Data. The three primary benchmarks (TempCompass, MVBench, NExT-QA) and the two extended benchmarks (Video-MME, LongVideoBench) are public. Our Stage A EST-SFT conversations are derived from public clip sources; we release the generation scripts along with the deduplicated conversation pool. The verifier's key-evidence frame-range annotations used in R_{ev} are released as a separate asset.

Verifier audit. The 200-sample human-labeled audit behind Figure 6(a) and the reward-hacking trace behind Figure 6(b) are released as CSVs alongside the corresponding training run IDs, so that the reported 0.93 verifier accuracy and the 2.3% upper bound on the hacking ratio can be recomputed.

Compute and seeds. Hardware, framework, and seed policy are documented in Section B.9. All headline RL rows use mean±std over 5 seeds drawn from {42, 1337, 2024, 2025, 7}; ablation and diagonal rows use the 3-seed subset {42, 1337, 2024}; the 1-seed rows are noted in their captions.

Scope of positive claims. Positive claims are restricted to the 3B–7B backbone regime and to the evaluation suite described in Section 4.1, plus the extended benchmarks in Table 2 and the cross-

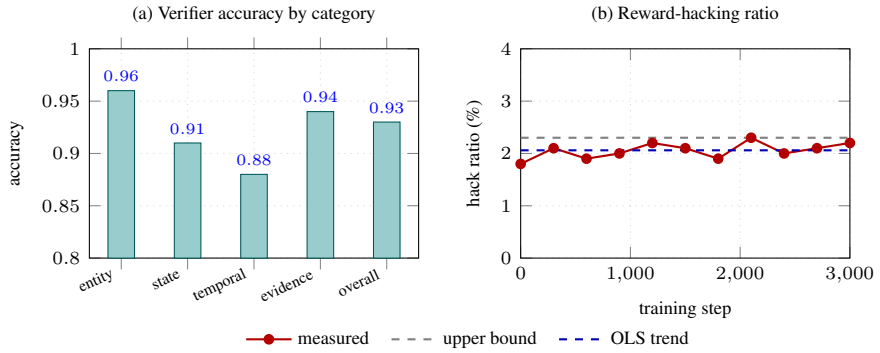

Figure 6: Verifier audit. Panel a reports verifier accuracy on human-labeled validation-pack samples stratified by structure category before annotation. Panel b reports the reward-hacking ratio, defined as the fraction of samples for which the verifier assigns a high score while human labels mark the answer as incorrect.

Table 12: Positioning of BPO along three design axes. Prior work covers individual axes; the bottom row is the joint combination this paper studies.

Method	**Optim. unit**	**Intervention**	**Advantage baseline**
LLaVA-Video / InternVideo (SFT) [9, 10]	response	none	—
Video-R1 [4]	response group	global, fixed	sampled-response baseline
TPO [5]	response pair	global, manipulated input	implicit (DPO)
CounterVid / MixDPO [14, 24]	response pair	global, synthesis-based	implicit (DPO)
BPO (ours)	**behavior pack**	**question-type conditioned**	**anchor-relative**

backbone rows in Table 10. Other scope limitations include verifier-schema coverage, ambiguous anchors, and synthetic-view artifacts.

F.4 Broader impacts

More evidence-sensitive video models could improve accessibility tools, education support, and safety-critical review workflows by reducing answers that ignore the relevant visual evidence. The same capability could also strengthen surveillance, automated moderation, or other high-stakes systems if deployed without consent, privacy safeguards, fairness audits, and human oversight. BPO is therefore best viewed as a reliability mechanism for audited settings rather than as a deployment guarantee.

F.5 Qualitative failure cases

We include this section to make the remaining error modes of BPO explicit. The cases below summarize validation-pack failures from the manual audit: the original clip is answered correctly, but one intervened view exposes a behavior that the contract does not fully repair.

- **Case F1: schema-out-of-scope question.** A counterfactual question of the form "what would have happened if X were absent?". Failure mode: only R_{ans} carries signal, the contract term degenerates, and the policy defaults to the anchor answer.
- **Case F2: ambiguous anchor.** A long-form clip in which two plausible reference segments exist, so the anchor view is not unique. Failure mode: the anchor-relative advantage receives an ambiguous signal and the policy oscillates across seeds.
- **Case F3: synthesis artifact.** A key-mask view leaves a visible seam that the policy picks up as a cue. Failure mode: key-sensitivity appears high for the wrong reason.

G Extended related work

This appendix expands the compact related-work summary in Section 2 into a per-axis discussion. We organize prior work along the three design axes that define BPO: the unit the loss optimizes, how

interventions are conditioned, and the form of the advantage baseline. A tabular summary appears in Table 12.

Video MLLM post-training. Strong video-language backbones such as LLaVA-Video [9], InternVideo [10], and Qwen2.5-VL [11] are typically post-trained with supervised fine-tuning on instruction-style question-answer pairs. Under this recipe, supervision is per-response: each answer is optimized against a single ground-truth target, usually on the original clip. Recent work adds structured intermediate reasoning traces [19] on top of SFT, but the optimization unit remains a single response. BPO instead optimizes joint outputs across counterfactual views and supervises behavior contracts that are invisible to a single-response loss.

Group-relative policy optimization for video. Following the success of group-relative policy optimization in language models [2, 3], Video-R1 [4] brings GRPO to video by augmenting each response group with shuffled-frame negatives under a global temporally-aware reward, encouraging the policy to be sensitive to ordering. The augmentation is applied uniformly across all questions and the reward is computed per response. BPO conditions the intervention set on question type, so order-sensitive and binding-sensitive questions receive different counterfactual contracts, and lifts reward computation from per-response to per-pack while replacing the batch-normalized pack baseline with an anchor-relative one.

Counterfactual and interventional supervision. Temporal Preference Optimization (TPO) [5] forms preference pairs from manipulated inputs (irrelevant clips, incomplete clips) and trains with DPO [1]; intervention behavior is captured pairwise. CounterVid [14] and MixDPO [24] synthesize counterfactual videos and build preference pairs to mitigate action and temporal hallucination. The pack-level contract reward of BPO captures multi-way behavior consistency (stability, sensitivity, abstention) jointly rather than as pairwise preferences, and the optimization unit and the reward share the same granularity, removing the mismatch between attribution and update that token-level shaping has to absorb.

Variance reduction in policy optimization. A long line of work in policy gradient methods replaces the naive Monte Carlo baseline with a learned value function or a sample-based control variate [25]. GRPO drops the value function entirely and normalizes within a sampled group of rollouts [2, 3], trading parameter overhead for empirical variance from small group sizes. The anchor-relative advantage is a structurally privileged per-prompt control variate: it uses the rollout of the same prompt under the identity view as a baseline. The batch-normalized pack alternative averages across prompts and is fragile when group sizes are small, which is the typical video post-training regime; the anchor-relative baseline is computed within the prompt and is grounded on the original clip, giving a strict variance reduction whenever the cross-view reward correlation exceeds an explicit threshold that depends only on pack size (Proposition 2; the threshold is about 0.64 for the $m = 4$ packs we use).

Object-centric and structured intermediate representations. A parallel line of work imposes object- or slot-level representations on the visual stream [26]. These approaches change the representation hypothesis space, which is complementary to the optimization hypothesis of BPO. Within our pipeline, structured outputs serve as a verifier interface, not as the locus of the contribution; the backbone is unchanged and the contribution lives in the post-training stage.

Tabular summary. Table 12 compares BPO to the methods discussed above along the three design axes. Prior work covers individual axes; BPO is the first system we are aware of to combine all three.

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 Section 1 state the paper's main contribution as Behavior Pack Optimization, together with the intended setting of video MLLMs post-training. The quantitative claims are supported by the controlled experiments in Section 4, and the scope is bounded by the limitations in Section 5 and the reproducibility/scope statement in Appendix F.3.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: The conclusion (Section 5) includes a dedicated Limitations paragraph on dependence on interventions and verifier reliability, and notes scope gaps such as open-ended generation, dense videos, and larger backbones.

   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: [Yes]

   Justification: The theoretical results are stated with assumptions in Section A, including the single-view reduction, anchor-relative variance reduction, and matched-budget consistency result. Complete proofs and the conditions under which they apply are provided in the same appendix and referenced from the method section.

   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: Section 4 specifies the benchmarks, metrics, baselines, controlled settings, rollout budgets, seed policy, and statistical reporting used for the main results. Sections B and F.3 further describe the training pipeline, counterfactual view construction, verifier, hardware and framework stack, and planned reproducibility artifacts.

   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: Benchmarks and backbone weights are public (Section 4.1). The supplemental material includes an anonymized code and configuration package with instructions to reproduce the main BPO pipeline; Sections B.9 and F.3 give the full environment summary.

   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: The experimental setup in Section 4.1 describes the backbones, benchmarks, metrics, controlled baselines, rollout budgets, and seed protocol. Section B provides the Stage-A and Stage-B implementation details, including view construction, verifier specification, training framework, hardware, and relevant hyperparameter choices.

   Guidelines:

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

7. **Experiment statistical significance**

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

   Answer: [Yes]

Justification: The main RL rows in Table 1 report mean and standard deviation over five seeds. Ablation and diagnostic rows state their seed policy in the captions and Appendix F.3. We avoid significance claims for single-seed diagnostics and use them only as mechanism checks.

Guidelines:

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

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: Section B.9 reports the hardware and framework stack, including the use of $8\times$ NVIDIA H20 GPUs. Table 6 reports rollout counts, pack sizes, relative FLOPs, and relative wall-clock costs for the Stage-B methods used in the main comparison.

   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 uses public benchmarks and model/code assets, and the paper does not report experiments involving private personal data, human subjects, or deployment in a high-risk decision setting. We have reviewed the NeurIPS Code of Ethics and are not aware of any deviation from it.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.

- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

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

    Answer: [Yes]

    Justification: Appendix F.4 discusses potential benefits for accessibility and reliability, possible risks around surveillance and automated moderation, privacy and fairness considerations, and limits of verifier-based evaluation without human oversight.

    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: [N/A]

    Justification: The paper does not introduce or release a new high-risk pretrained generative model, image generator, or scraped internet dataset.

    Guidelines:

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

12. **Licenses for existing assets**

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

Answer: [Yes]

Justification: The paper cites the original works for the existing datasets, models, and tools used in the experiments. These assets are publicly available, and our use follows their open-source licenses or public terms of use.

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: Section F.3 documents the derived assets released with the work, including Stage-A EST-SFT conversations, key-evidence frame-range annotations for the verifier, and verifier-audit CSVs. These assets are derived from public clip sources and are accompanied by generation scripts, run identifiers, and configuration files for reproducibility.

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 report crowdsourcing experiments or research with human subjects. The verifier and question-type audits use annotations by paper authors, not crowd workers or external participants.

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: The work does not involve crowdsourcing or human-subject experiments, so IRB approval or an equivalent human-subject review is not applicable. The paper-author-labeled audits evaluate model/verifier behavior on validation samples rather than intervening with or collecting data from study participants.

    Guidelines:

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

16. **Declaration of LLM usage**

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

    Answer: [Yes]

    Justification: LLMs are part of our experiments (video MLLM backbones and the LLM judge in the verifier, described in Sections B.4, 3 and 4.1). In addition, we used LLMs only to help polish the manuscript.

    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.