LEARNING SPATIOTEMPORAL SENSITIVITY IN VIDEO LLMS VIA COUNTERFACTUAL REINFORCEMENT LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Video large language models (Video LLMs) can achieve strong video-QA accuracy without reliably tracking spatiotemporal dynamics. A model may answer a motion question from static cues, for example, and give the same prediction even after the underlying motion is reversed. Correctness-based reinforcement learning does not directly address this problem because it rewards the final answer without requiring the policy to respond to task-relevant video dynamics. We propose **Counterfactual Relational Policy Optimization (CRPO)**, which explicitly trains Video LLMs to respond to controlled changes in the visual input. For each training example, CRPO constructs a counterfactual video, such as a horizontally flipped or temporally reversed version of the original, and jointly optimizes rollouts from both videos under a shared policy. Factual supervision anchors *what the model should answer*, while counterfactual supervision constrains *when that answer should change*: predictions should change when an intervention alters task-relevant dynamics and remain stable when the queried property is preserved. This coupling provides a direct behavioral learning signal without requiring ground-truth labels for transformed videos or annotated spatiotemporal reasoning traces, while discouraging indiscriminate answer changes. To evaluate this property, we introduce **DyBench**, a paired counterfactual benchmark with 3,014 videos and a strict pair-accuracy metric. Across paired spatiotemporal evaluations and standard video benchmarks, CRPO improves sensitivity to motion and temporal changes while improving performance on general video understanding. The gains also extend to segment reordering, a transformation never used during training, suggesting that CRPO learns sensitivity to video dynamics beyond the training interventions.

1 INTRODUCTION

Video large language models (Video LLMs) (Zhang et al., 2024; Bai et al., 2025a; Wang et al., 2025d; Pichai et al., 2025) have achieved strong results on video understanding benchmarks (Fu et al., 2025; Li et al., 2024b). Yet high answer accuracy does not necessarily imply that a model tracks spatiotemporal dynamics. Recent studies show that Video LLMs can often solve video questions using static shortcuts, such as single-frame cues or language priors, instead of following the underlying dynamics (Lei et al., 2023; Zohar et al., 2025; Li et al., 2026; Krojer et al., 2025). Figure 1 illustrates this gap: Qwen3-VL (Bai et al., 2025a) can recognize a static property of a scene, but fails to track movement direction and gives the same prediction even after the video is temporally reversed. More generally, performance degrades on questions that require motion direction, event order, and other spatiotemporal information. These observations expose a capability that answer accuracy alone does not measure: a model should not only answer correctly, but also respond appropriately when the relevant video dynamics change.

This distinction is especially important in reinforcement-learning post-training. RL has become an effective way to improve reasoning in LLMs (Guo et al., 2025; Yu et al., 2025) and Video LLMs (Feng et al., 2025a; Wang et al., 2025b; Li et al., 2025b), but standard objectives primarily reward whether the final answer is correct. If a training question can be solved from a salient frame or a language prior, a shortcut policy can receive the same reward as a policy that genuinely tracks the video dynamics. Correctness-based RL therefore provides no explicit incentive for the prediction to respond when

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

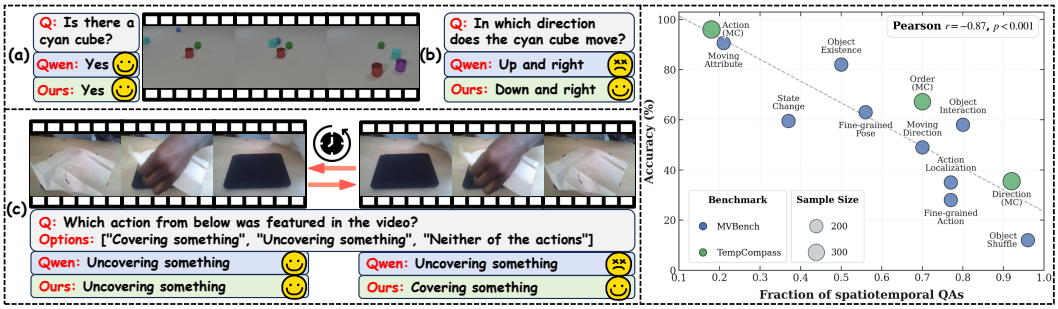

Figure 1: **High video-QA accuracy does not necessarily imply sensitivity to video dynamics. Left:** The model **(a)** answers a static question correctly, but **(b)** fails on a spatiotemporal question; **(c)** it gives the same prediction to a video and its temporal reversal. **Right:** Across MVBench and TempCompass sub-tasks, accuracy drops as the fraction of spatiotemporal questions increases.

motion, temporal order, or other task-relevant evidence changes. A direct alternative would be to annotate which frames, motions, or events support each answer (Yuan et al., 2025; Shi et al., 2026), but such process-level supervision is expensive and difficult to scale.

Our key observation is that *controlled changes to the video provide a much simpler supervision signal*. Consider a question asking whether an object moves left or right. If an appropriate transformation changes the direction of motion, a model that uses this evidence should also change its answer. In contrast, if the question asks about the object's color or presence, the same transformation should leave the answer unchanged. This yields a simple principle: *predictions should change when an intervention changes task-relevant evidence, and remain stable when it does not*. We refer to this property as **spatiotemporal sensitivity**. Importantly, this principle specifies only the expected *relationship* between the predictions before and after the intervention. It therefore provides useful supervision even when the exact ground-truth answer to the transformed video is unavailable.

Based on this principle, we propose **Counterfactual Relational Policy Optimization (CRPO)**. For each video question, CRPO constructs a controlled counterfactual video using a task-appropriate spatial or temporal transformation, and generates rollouts from both the original and counterfactual inputs. The original branch is supervised by the ground-truth answer, while the counterfactual branch is trained according to how the answer should respond to the transformation. A **Counterfactual Relation Reward (CRR)** couples the two branches, encouraging the model to preserve its prediction when the queried property is unchanged and to alter it when task-relevant dynamics change. Since both branches update a shared policy, ground-truth supervision on the factual branch anchors answer correctness, while relational supervision constrains when predictions should change; Static examples additionally require predictions to remain stable when the queried property is preserved. Together, these signals discourage indiscriminate answer changes without requiring transformed-video labels.

CRPO also differs from prior perturbation-based video RL methods such as T-GRPO (Feng et al., 2025a) and ArrowRL (Xue et al., 2025). These methods use perturbed videos primarily to shape rewards for rollouts from the original video, whereas CRPO directly optimizes rollouts from both factual and counterfactual branches. We further identify an optimization issue in such reward-shaping approaches: when an auxiliary reward acts as a common rescaling within a GRPO rollout group, group-wise advantage normalization can cancel its effect on the policy gradient. By contrast, CRPO obtains rollout-level behavioral variation directly from the counterfactual branch. We analyze this effect in Section 3.4.

We evaluate CRPO on paired spatiotemporal diagnostics and standard video benchmarks. Alongside existing benchmarks, we introduce **DyBench**, a paired diagnostic benchmark in which closely matched videos differ in their dynamics and therefore require different answers. Its strict pair accuracy requires both sides of a pair to be answered correctly, making fixed-answer shortcuts ineffective. Across Qwen3-VL-4B and Qwen3-VL-8B, CRPO consistently improves paired spatiotemporal evaluations while preserving performance on original videos and general video understanding. The gains also extend to Event Sequence examples constructed by segment reordering, a transformation never used during CRPO training. These results suggest that CRPO learns a broader sensitivity to video dynamics rather than simply fitting the training interventions.

Our contributions are threefold:

- We formulate **spatiotemporal sensitivity** as a counterfactual relational learning problem: predictions should change when a controlled intervention alters task-relevant dynamics and remain stable when the queried property is preserved.
- We propose **CRPO**, a dual-branch RL framework that combines ground-truth supervision on factual videos with relational supervision on counterfactual videos under a shared policy, without requiring ground-truth answers for transformed videos or process-level spatiotemporal annotations.
- We introduce the paired **DyBench** diagnostic and conduct controlled and out-of-distribution evaluations showing that CRPO improves sensitivity to spatiotemporal changes while preserving general video understanding.

2 Related Work

Video Large Language Models. The rapid progress of LLMs (Pichai et al., 2025; Singh et al., 2025) has driven substantial advances in video understanding. Recent Video LLMs typically extend the image-language instruction-tuning paradigm (Liu et al., 2023) to video by bridging video and text modalities through lightweight adapters or projectors, and by training on large-scale video-caption and video instruction-following data (Li et al., 2024a; Zhang et al., 2025; 2024). More recent open models (Cho et al., 2025; Clark et al., 2026; Wang et al., 2025d; Bai et al., 2025a) further strengthen multimodal reasoning, grounding, and long-context modeling through better data and training pipelines. Some work improves long-video understanding through memory, compression, or token reduction (Song et al., 2024; Shu et al., 2025; Shen et al., 2024). Despite these advances, strong benchmark performance does not necessarily imply genuine spatiotemporal understanding, as many Video LLMs can still rely on static shortcuts such as single-frame cues or language priors (Zohar et al., 2025; Krojer et al., 2025) rather than truly modeling how events unfold over time.

RL for Video LLMs. Recent work has explored reinforcement learning for Video LLMs. GRPO (Guo et al., 2025) has become a common foundation for RL-based post-training, motivating studies on data selection, reward design, optimization stability, and credit assignment (Li et al., 2025b; Wang et al., 2025b; Park et al., 2025; Feng et al., 2025b; Liu et al., 2026; Wang et al., 2026). These works show that video RL depends critically on how reward signals are designed. Several recent methods further introduce video variations into RL. Video-R1 (Feng et al., 2025a) (T-GRPO) shuffles frames and rewards original-video rollouts when they outperform shuffled-video rollouts, while ArrowRL (Xue et al., 2025) uses temporal reversal to penalize original-video rollouts that mirror reversed-video responses. These perturbations provide useful contrastive signals, but the perturbed videos are used only for reward shaping. STRIVE (Bahrami et al., 2026) groups rollouts jointly over textual outputs and spatiotemporal visual variants to improve exploration. CRPO follows the direction of using video variations, but treats the counterfactual video as a trainable branch: both original and counterfactual branches generate rollouts and contribute policy gradients. The Counterfactual Relation Reward then links the two branches by rewarding answer changes for dynamic questions and answer preservation for static questions, yielding a simple and targeted signal for spatiotemporal sensitivity.

Video Understanding Benchmarks. Standard benchmarks (Fu et al., 2025; Li et al., 2024b; Wang et al., 2025c; Zhao et al., 2025) cover broad video understanding abilities, but they do not explicitly diagnose shortcut-based success. Recent diagnostic benchmarks aim to close this gap. TempCompass (Liu et al., 2024) probes temporal perception with conflicting videos matched in static content. MHBench (Kong et al., 2025) uses adversarial triplets to evaluate motion hallucination and test whether Video LLMs rely on static appearance instead of true motion perception. MVP (Krojer et al., 2025) uses minimal-change video pairs with identical questions and opposite answers to reduce shortcut-based score inflation. TimeBlind (Li et al., 2026) is closest to our setting, using minimal video pairs and paired evaluation to isolate temporal understanding. Our proposed *DyBench* follows this line, but frames paired evaluation through counterfactual video pairs. It focuses on three spatiotemporal tasks, namely reversible dynamics, moving direction, and event sequence, and uses *pair accuracy* to distinguish shortcut success from genuine spatiotemporal understanding.

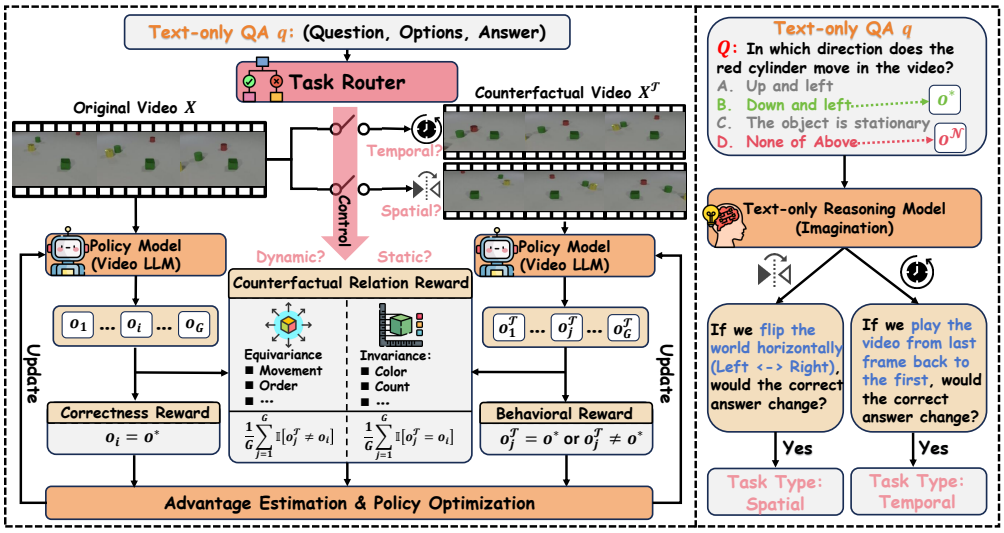

Figure 2: **Overview of CRPO. Left:** Given a video question, the Task Router selects a counterfactual transformation $\mathcal{T}$ (horizontal flip or temporal reversal). The original branch and the counterfactual branch each generate G rollouts. These rollouts are scored by branch-specific correctness or behavioral rewards, format rewards, and the Counterfactual Relation Reward (CRR), and are then used for advantage estimation and policy optimization. **Right:** The Task Router is implemented as a text-only reasoning model that answers two hypothetical transformation questions to predict whether the correct answer would change under horizontal flip or temporal reversal.

3 Counterfactual Relational Policy Optimization

Standard GRPO (Guo et al., 2025) optimizes whether a sampled response to a video question is correct, but does not require the prediction to respond when the underlying video dynamics change. CRPO introduces this missing supervision by jointly training on a factual video and a controlled counterfactual version of it. Given a training example (X, Q, y), where X is the video, Q is the question, and y is the ground-truth answer, we construct a counterfactual video $X^{\mathcal{T}}$ and generate rollout groups from both (X, Q) and $(X^{\mathcal{T}}, Q)$ using the same policy. The factual branch provides supervision on *what to answer*, while the counterfactual branch provides supervision on *when the answer should change*. Figure 2 gives an overview of the framework.

3.1 Task-Aware Counterfactual Construction

The effect of a visual transformation depends on what the question asks. For example, horizontally flipping a video can change the answer to a left–right motion question while leaving object color or identity unchanged. Similarly, temporal reversal can change event order or reversible actions such as opening and closing, while preserving static scene content.

As shown in the right panel of Figure 2, we determine this dependency offline using a text-only **Task Router**. Given only the question, answer options, and ground-truth answer, the router considers two hypothetical interventions: whether the correct answer would change if the video were horizontally flipped, and whether it would change if the video were played in reverse. The two decisions categorize the question as Spatial, Temporal, Spatiotemporal, or Static. Spatial questions use horizontal flipping, Temporal questions use temporal reversal, and Spatiotemporal questions randomly use one of the two transformations. For Static questions, either transformation can be used because the queried property should remain unchanged. The router is executed once before RL training and introduces no additional inference cost during optimization. Its full prompt, classification statistics, and human validation are provided in Appendix C.

We refer to Spatial, Temporal, and Spatiotemporal questions collectively as *dynamic*. For a dynamic question, the selected intervention changes task-relevant evidence and the answer is therefore expected to change. For a Static question, the queried property is preserved and the answer should remain

unchanged. This gives CRPO a counterfactual supervision signal without requiring a new ground-truth label for $X^{\mathcal{T}}$.

3.2 Counterfactual Relational Reward

For each training example (X, Q, y), CRPO samples G factual rollouts $\{o_i\}_{i=1}^{G}$ from the original video and G counterfactual rollouts $\{\tilde{o}_j\}_{j=1}^{G}$ from the transformed video $X^{\mathcal{T}}$. Let a_i and $\tilde{a}_j$ denote their extracted final answers. The factual branch follows standard outcome supervision, with $c_i = \mathbb{I}[a_i = y]$ indicating whether a rollout is correct. The counterfactual branch instead evaluates whether the prediction responds to the intervention as expected: for dynamic questions, $b_j = \mathbb{I}[\tilde{a}_j \neq y]$, while for Static questions, $b_j = \mathbb{I}[\tilde{a}_j = y]$. We refer to this local counterfactual term as the behavioral reward, $R_{\text{behave}}(\tilde{o}_j) = b_j$. We denote their group averages by $\bar{c} = \frac{1}{G}\sum_i c_i$ and $\bar{b} = \frac{1}{G}\sum_j b_j$.

CRPO couples factual correctness and counterfactual behavior through a symmetric relational reward:

$$R_i^{\text{F}} = c_i + \lambda_z c_i \bar{b} + R_{\text{fmt}}(o_i), \tag{1}$$

$$R_j^{\text{CF}} = w_{\text{aug}}\left(b_j + \lambda_z b_j \bar{c} + R_{\text{fmt}}(\tilde{o}_j)\right), \tag{2}$$

where λ_z is λ_d for dynamic questions and λ_s for Static questions, and w_{aug} controls the overall contribution of the counterfactual branch. R_{fmt} is the standard format reward used to enforce the required response structure. The first term in each branch provides its local supervision, with factual correctness used for the original video and the expected behavioral response used for the counterfactual video, while the second term is the **Counterfactual Relation Reward (CRR)**. A correct factual rollout receives additional reward when the counterfactual branch consistently responds to the intervention, and conversely, a counterfactual rollout is reinforced more strongly when the factual branch is reliably correct. CRR therefore encourages the two predictions to be jointly consistent with the underlying intervention rather than optimizing the two branches independently.

This decomposition reflects the complementary roles of the two supervision signals. Factual supervision anchors *what the model should answer*, whereas counterfactual supervision constrains *when that answer should change*. For dynamic questions, the relational signal specifies that the answer should change after the intervention, but does not determine the exact counterfactual answer, making it weaker than supervision from exact counterfactual labels. CRPO therefore does not optimize this signal in isolation. Because factual and counterfactual rollouts update the same policy, ground-truth supervision anchors factual correctness, while relational supervision encourages selective responses to task-relevant changes; Static examples constrain the policy to preserve predictions when the queried property remains unchanged. In this way, counterfactual relations serve as a behavioral constraint on a factually grounded policy, without requiring ground-truth labels for transformed videos.

A final practical issue arises for multiple-choice questions. After a spatial or temporal transformation, the correct counterfactual answer may no longer appear among the original answer choices. We therefore append a *None of the above* option $o^{\mathcal{N}}$ (Null Option) during training. It is always incorrect for the factual video, where the original ground-truth answer remains available, but allows the counterfactual branch to represent cases in which the transformation moves the valid answer outside the original choice set. We analyze its selection behavior during training in Appendix D.1.

3.3 Dual-Branch Policy Optimization

Both factual and counterfactual rollouts directly optimize the same policy π_θ. Following GRPO, rewards are centered within each rollout group. CRPO centers the factual and counterfactual rewards separately, but normalizes both groups using a shared standard deviation computed over the centered rewards from the two branches. This preserves the relative ordering of rollouts within each branch while placing their advantages on a common optimization scale. We denote the resulting advantages by $\widehat{A}_k^{\text{F}}$ and $\widehat{A}_k^{\text{CF}}$.

Let $b \in \{\text{F}, \text{CF}\}$ index the factual and counterfactual branches, and let ρ_k^b denote the corresponding importance ratio between the current and old policies. CRPO optimizes

$$\mathcal{J}_{\text{CRPO}}(\theta) = \sum_{b\in\{\text{F},\text{CF}\}} \frac{1}{G}\sum_{k=1}^{G} \min\Big(\rho_k^b \widehat{A}_k^b, \text{clip}(\rho_k^b, 1-\epsilon, 1+\epsilon)\widehat{A}_k^b\Big), \tag{3}$$

with standard GRPO KL regularization applied to both branches. Exact normalization details and the KL setting are provided in Appendix D.2.

The key distinction from prior perturbation-based methods lies in how the transformed video participates in optimization. T-GRPO (Feng et al., 2025a) and ArrowRL (Xue et al., 2025) primarily use perturbed videos to modify the rewards of rollouts sampled from the original input. CRPO instead treats the counterfactual video as an explicit optimization branch: its rollouts receive their own advantages and directly contribute policy gradients to the shared model. Thus, counterfactual information enters policy learning through the rollouts themselves rather than only through an auxiliary reward on factual responses. As we show next, this distinction also avoids an important limitation of perturbation-based reward shaping under group-wise advantage normalization.

3.4 Reward Cancellation under Group-Wise Advantage Normalization

A key limitation of perturbation-based reward shaping is that its auxiliary signal can be suppressed by the group-wise advantage normalization used in GRPO. This issue is particularly relevant to methods such as T-GRPO (Feng et al., 2025a) and ArrowRL (Xue et al., 2025), where information from the perturbed video is injected into the rewards of factual rollouts rather than optimized through an independent rollout branch.

Consider the case in which the perturbation-derived signal assigns the same bonus C to every correct rollout in a group. For binary correctness $c_i \in \{0, 1\}$, with group mean $\bar{c}$ and standard deviation σ_c, the normalized advantage becomes

$$\widehat{A}_i = \frac{(1 + C)(c_i - \bar{c})}{(1 + C)\sigma_c + \epsilon} \approx \frac{c_i - \bar{c}}{\sigma_c}. \tag{4}$$

The common factor $(1 + C)$ therefore cancels under normalization. Although the auxiliary signal changes the numerical reward, it contributes little or no additional relative advantage in this case, and hence provides essentially no new policy-gradient signal. This reveals an important limitation of injecting perturbation information only through group-level reward shaping.

CRPO avoids this cancellation by optimizing counterfactual rollouts directly. Because the counterfactual responses are independently sampled, their behavioral rewards vary across rollouts according to whether each prediction responds appropriately to the intervention. These rollout-level differences survive group-wise normalization and produce counterfactual advantages that directly update the shared policy. Thus, counterfactual information enters CRPO through the policy gradient itself, rather than only through a potentially normalized-away reward modifier. The full derivation and additional analysis are provided in Appendix D.3.

4 DyBench: A Paired Benchmark for Spatiotemporal Sensitivity

Existing video benchmarks are dominated by questions whose answers can often be inferred from a single frame or a language prior (Krojer et al., 2025; Li et al., 2026), leaving open the question of whether a Video LLM truly tracks spatiotemporal dynamics. To diagnose this directly, we construct **DyBench**, a paired counterfactual benchmark of 1,507 video pairs (3,014 videos) targeting motion direction, event order, and the arrow of time, three aspects of video understanding that are difficult to resolve from static cues alone.

Counterfactual pairs and pair accuracy. Each DyBench item is a *counterfactual pair* (v, v') rather than a single video. The two videos share the same underlying scene and objects, ask the same multiple-choice question, and use the same semantic answer choices, while differing along a controlled spatiotemporal axis that changes the correct answer. We report **pair accuracy (P-Acc)**, which counts a pair as correct only when the model answers both v and v' correctly. By construction, a model that returns the same answer despite the changed dynamics cannot succeed on the pair. Compared with standard per-video accuracy (Acc), P-Acc therefore provides a stricter measure of whether model predictions actually respond to the relevant spatiotemporal change.

Three sub-tasks. DyBench covers three complementary forms of dynamic reasoning. **(1) Reversible Dynamics** asks whether a process unfolds in one temporal direction or the other, such as

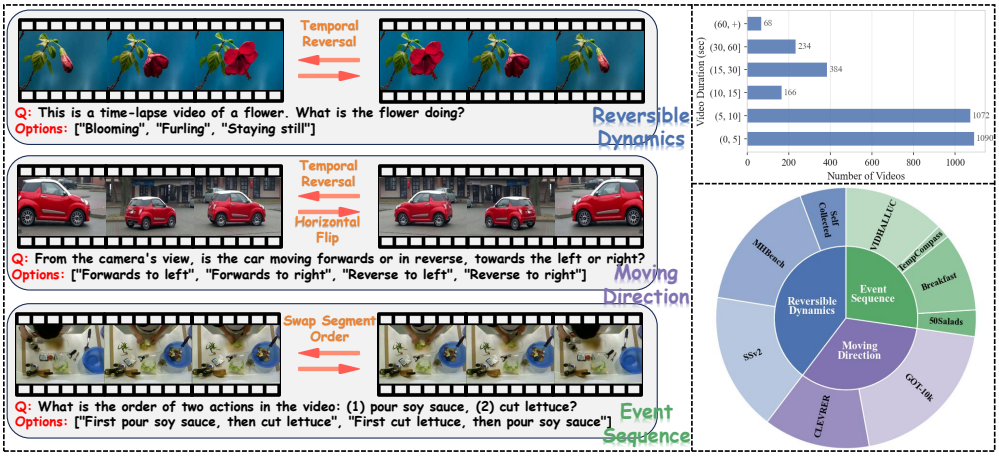

Figure 3: **Overview of DyBench.** DyBench contains 3,014 videos forming 1,507 counterfactual pairs across three sub-tasks: Reversible Dynamics, Moving Direction, and Event Sequence. The upper-right panel shows the distribution of video durations, while the lower-right panel shows the distribution of video sources.

opening versus *closing* or *blooming* versus *closing*, and constructs pairs through temporal reversal. **(2) Moving Direction** asks in which direction an object or actor moves, with paired videos constructed through horizontal flipping and, when appropriate, temporal reversal. **(3) Event Sequence** asks about the order of two events and constructs pairs by concatenating the same action segments in opposite orders. As summarized in Figure 3, DyBench draws from diverse video sources and covers a range of short-video durations. Full construction details, source datasets, quality control, shortcut-isolation analysis, and per-sub-task results are provided in Appendix A.

5 Experiments

5.1 Experimental Setup

Benchmarks. We evaluate on five benchmarks. DyBench (Section 4) and TimeBlind (Li et al., 2026) are paired benchmarks for which we report both standard accuracy and a strict paired metric: pair accuracy (P-Acc) for DyBench and instance accuracy (I-Acc) for TimeBlind. For DyBench, we additionally report **original accuracy (O-Acc)**, which evaluates only the original, untransformed videos in each pair. The other three benchmarks, TempCompass (Liu et al., 2024), VideoMME (Fu et al., 2025), and MVBench (Li et al., 2024b), use standard accuracy. All evaluations are conducted with VLMEvalKit (Duan et al., 2024) using 32 uniformly sampled frames.

Baselines. We compare CRPO against two proprietary models, GPT-5.1 (Singh et al., 2025) and Gemini-3.1-Pro (Pichai et al., 2025), and several open-source Video LLMs, including LLaVA-OneVision-7B (Li et al., 2024a), InternVL3-8B (Wang et al., 2025d), Qwen2.5-VL-7B (Bai et al., 2025b), and ArrowRL*, the officially released checkpoint of Xue et al. (2025). We further compare CRPO with GRPO (Guo et al., 2025), T-GRPO (Feng et al., 2025a), and ArrowRL (Xue et al., 2025) under controlled training settings. All RL post-training methods use the same backbone, training corpus, and shared hyperparameters. Further details are provided in Appendix B.3.

Training Details. We instantiate CRPO on Qwen3-VL-4B-Instruct and Qwen3-VL-8B-Instruct (Bai et al., 2025a), and follow the same training corpus and RL hyperparameters as VideoAuto-R1 (Liu et al., 2026). The corpus mixes text, image, and video QA; the dual-branch counterfactual framework is applied only to the video subset, while text and image samples are trained with standard single-branch GRPO. The video subset contains 21,165 samples and is processed with G=8 rollouts per branch, 32 uniformly sampled frames per video, and a learning rate of 1×10^{-6}. We set $\lambda_d = \lambda_s = 0.3$ and $w_{\text{aug}} = 0.5$. The Task Router is run once offline using DeepSeek-R1. Training is conducted on 32 NVIDIA H20 GPUs. Full data composition, reward definitions, hyperparameters, and Task Router details are provided in Appendices B.1, B.2, and C.

Table 1: **Main results on five video benchmarks.** O-Acc denotes accuracy on only the original, untransformed videos of DyBench, while P-Acc and I-Acc are the strict paired metrics of DyBench and TimeBlind, respectively. Values marked with ↑ denote absolute percentage-point improvements of CRPO over the corresponding Qwen3-VL base model.

Models	Spatiotemporal						General	
	DyBench			TimeBlind		TempCompass	VideoMME	MVBench
	Acc	O-Acc	P-Acc	Acc	I-Acc	Acc	Acc	Acc
Proprietary Models								
GPT-5.1 (Singh et al., 2025)	63.7	76.7	44.9	67.3	27.0	76.4	72.9	62.9
Gemini-3.1-Pro (Pichai et al., 2025)	82.2	88.3	71.7	77.2	45.5	74.5	74.3	73.6
Open-Source Models								
LLaVA-OV-7B (Li et al., 2024a)	48.4	59.1	19.6	56.5	7.8	60.5	57.6	54.7
InternVL3-8B (Wang et al., 2025d)	69.5	79.0	50.2	63.0	18.3	70.5	65.6	74.3
Qwen2.5-VL-7B (Bai et al., 2025b)	64.5	76.0	44.1	65.0	22.5	70.9	60.3	67.1
ArrowRL* (Xue et al., 2025)	67.5	78.8	50.1	65.0	20.2	71.7	60.1	66.2
RL Post-Training of Qwen3-VL-4B								
Qwen3-VL-4B (base) (Bai et al., 2025a)	65.1	76.2	45.4	66.2	26.5	71.6	62.2	67.5
+ GRPO (Guo et al., 2025)	66.6	78.0	48.2	67.8	28.0	72.2	62.1	68.7
+ T-GRPO (Feng et al., 2025a)	68.6	78.5	49.8	67.9	29.0	73.5	**63.4**	**70.3**
+ ArrowRL (Xue et al., 2025)	66.7	77.6	47.5	67.5	26.8	71.6	62.2	69.9
+ **CRPO (Ours)**	**70.3**↑5.2	**79.5**↑3.3	**54.8**↑9.4	**69.8**↑3.6	**31.7**↑5.2	**74.2**↑2.6	63.0↑0.8	68.6↑1.1
RL Post-Training of Qwen3-VL-8B								
Qwen3-VL-8B (base) (Bai et al., 2025a)	68.2	78.9	50.4	67.8	27.8	75.1	64.9	68.7
+ GRPO (Guo et al., 2025)	69.4	79.9	52.4	69.8	30.5	75.5	64.5	69.1
+ T-GRPO (Feng et al., 2025a)	70.9	80.8	54.5	70.3	32.5	75.4	65.0	69.5
+ ArrowRL (Xue et al., 2025)	69.6	80.3	52.2	69.3	29.3	74.9	65.1	69.7
+ **CRPO (Ours)**	**72.5**↑4.3	**81.6**↑2.7	**58.1**↑7.7	**71.7**↑3.9	**36.0**↑8.2	**77.4**↑2.3	**65.6**↑0.7	**69.7**↑1.0

5.2 Main Results

Table 1 summarizes the results across the five benchmarks. **CRPO consistently improves spatiotemporal sensitivity.** On Qwen3-VL-4B, CRPO improves DyBench P-Acc by $+9.4$ and TimeBlind I-Acc by $+5.2$ over the base model, and further outperforms T-GRPO by $+5.0$ and $+2.7$ points, respectively. The same trend holds on Qwen3-VL-8B, where CRPO exceeds T-GRPO by $+3.6$ P-Acc on DyBench and $+3.5$ I-Acc on TimeBlind. These gains are larger on the strict paired metrics than on standard accuracy, indicating that CRPO particularly improves the model's response to changes in video dynamics. Importantly, this does not come from specializing to transformed test inputs: on the original, untransformed DyBench videos, CRPO improves O-Acc from 76.2 to 79.5 for 4B and from 78.9 to 81.6 for 8B. The improvement also extends to Event Sequence, which uses segment reordering rather than the transformations seen during training; full per-sub-task results are provided in Appendix A.5, with paired qualitative examples in Appendix F. **CRPO preserves general video understanding and scales to the larger backbone.** CRPO improves over the corresponding base model on TempCompass, VideoMME, and MVBench for both 4B and 8B. On the 4B backbone, T-GRPO remains stronger on VideoMME and MVBench, while CRPO achieves substantially better paired spatiotemporal performance. On the 8B backbone, CRPO obtains the best RL result on VideoMME and ties the best result on MVBench, while improving DyBench P-Acc and TimeBlind I-Acc by $+7.7$ and $+8.2$ over the base model. Overall, CRPO improves sensitivity to spatiotemporal changes while maintaining the model's general video-understanding capability.

5.3 More Analysis

Controlled ablations. Table 2 shows that CRPO's gains cannot be explained by additional data or rollout computation alone. GRPO (G$\times$2) remains 4.6 points below CRPO on DyBench P-Acc and 2.7 points below on TimeBlind I-Acc, while GRPO (Data$\times$2) provides little improvement over standard GRPO. Removing the counterfactual relation reward (CRR) degrades paired performance, and removing the counterfactual branch causes the largest drop in DyBench P-Acc, from 54.8 to 48.8. Both counterfactual transformations contribute, with temporal reversal having a larger effect than spatial flipping, while removing the Null Option also causes substantial degradation. Additional diagnostics show that CRPO remains robust to moderate Task Router errors and that the Null Option is used selectively rather than collapsing across branches; see Appendices C and D.1.

Training-curve analysis. Figure 4 compares the training dynamics of CRPO, GRPO, T-GRPO, and ArrowRL. Three observations stand out. **(a) Correctness reward.** GRPO, T-GRPO, and ArrowRL rapidly increase factual correctness, reaching approximately 0.70 within the first 20% of training,

Table 2: **Ablation on Qwen3-VL-4B.** Top: controls for rollout computation and data exposure together with controlled RL baselines. Bottom: removing individual components of CRPO. The "w/o Counterfactual Branch" setting removes direct policy updates from the counterfactual branch while retaining its rollouts for the factual-branch relational reward.

RL Method	DyBench			TimeBlind		TempCompass	VideoMME	MVBench
	Acc	O-Acc	P-Acc	Acc	I-Acc	Acc	Acc	Acc
Baseline								
Qwen3-VL-4B (no RL)	65.1	76.2	45.4	66.2	26.5	71.6	62.2	67.5
Data / Compute Controls and RL Baselines								
GRPO	66.6	78.0	48.2	67.8	28.0	72.2	62.1	68.7
GRPO (G×2)	68.5	78.6	50.2	68.5	29.0	74.0	62.3	70.2
GRPO (Data×2)	66.5	78.0	48.4	67.4	27.3	73.0	62.0	69.8
T-GRPO	68.6	78.5	49.8	67.9	29.0	73.5	**63.4**	**70.3**
ArrowRL	66.7	77.6	47.5	67.5	26.8	71.6	62.2	69.9
CRPO (Ours)	**70.3**	**79.5**	**54.8**	**69.8**	**31.7**	**74.2**	63.0	68.6
Component Ablations								
w/o Spatial Flip	69.5	79.0	53.2	69.2	31.4	74.0	62.8	69.0
w/o Temporal Reversal	68.9	78.8	50.7	68.3	29.6	73.1	62.3	68.8
w/o CRR	69.3	78.9	51.2	68.8	30.5	73.1	63.0	67.5
w/o Counterfactual Branch	67.5	78.5	48.8	68.0	29.4	72.5	62.8	68.2
w/o Null Option	65.9	76.6	50.1	68.1	28.5	72.2	62.1	63.7

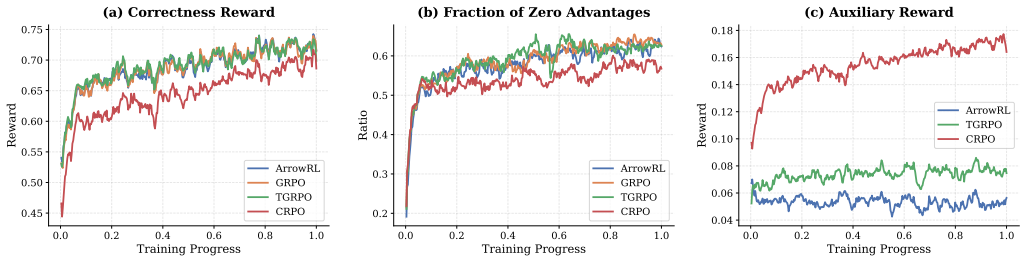

Figure 4: **Training dynamics of CRPO vs. RL baselines.** (a) factual correctness reward, (b) fraction of zero-advantage rollout groups, and (c) auxiliary or relational reward.

whereas CRPO rises more gradually before converging to a comparable final level, indicating that its gains do not come from optimizing factual correctness more aggressively. **(b) Fraction of zero advantages.** CRPO consistently exhibits fewer rollout groups with all-equal rewards and hence zero advantages, as the independently sampled counterfactual responses introduce additional rollout-level reward variation even when the factual branch is already highly accurate, providing a denser optimization signal throughout training. **(c) Auxiliary reward.** The perturbation-derived rewards of T-GRPO and ArrowRL remain low and relatively flat, consistent with the normalization effect analyzed in Section 3.4, whereas CRPO's relational reward increases steadily, indicating that the policy progressively learns to satisfy the desired relation between factual and counterfactual predictions. Together, these dynamics show that directly optimizing counterfactual rollouts provides a more persistent learning signal than correctness-only RL or perturbation-based reward shaping.

6 Conclusion

We present **CRPO**, a dual-branch reinforcement-learning framework for training Video LLMs to be sensitive to task-relevant spatiotemporal changes. CRPO combines factual and counterfactual supervision under a shared policy: factual supervision anchors what the model should answer, while counterfactual supervision constrains when that answer should change. By directly optimizing counterfactual rollouts, CRPO provides a direct spatiotemporal learning signal while avoiding the normalization-induced limitation of perturbation-based reward shaping. We also introduce **DyBench**, a paired benchmark with strict pair accuracy for measuring this behavior. Experiments show that CRPO improves spatiotemporal sensitivity and video understanding, with gains extending to an unseen transformation. Together, these results show that controlled counterfactual relations provide a simple but practical way to ground Video LLM predictions in visual dynamics.

AI USE STATEMENT

In this work, we used generative AI tools as a component of the proposed method and as a general-purpose writing aid. Specifically, DeepSeek-R1 is used offline as the Task Router (Section 3.1, Appendix C) to classify training questions according to their expected response to spatial and temporal transformations. The reliability of this component is evaluated through a human audit reported in Appendix C. We additionally used large language models to polish the wording of the manuscript and to assist with routine code and plotting scripts. All AI-assisted content was reviewed by the authors, who take full responsibility for the final manuscript and released artifacts.

ETHICS STATEMENT

This work does not involve human subjects, personally identifying information, or sensitive attributes. DyBench is constructed from existing research video datasets (Something-Something-V2, MHBench, GOT-10k, CLEVRER, Breakfast, 50Salads, VIDHALLUC, and TempCompass) together with web-sourced clips. We use these materials for research evaluation and follow the access and usage terms of the source datasets. The counterfactual transformations used in DyBench, including horizontal flipping, temporal reversal, and segment reordering, are geometric or temporal operations that do not synthesize new depicted individuals. Improving the spatiotemporal sensitivity of Video LLMs has the same broad dual-use considerations as video understanding more generally. Limitations and intended applications are discussed in the appendix. We declare no conflicts of interest.

REPRODUCIBILITY STATEMENT

The CRPO objective, Counterfactual Relation Reward, counterfactual construction, and dual-branch optimization are specified in Section 3. The training-data composition and standard reward definitions are provided in Appendix B.1, the full training configuration in Appendix B, and the Task Router prompt and validation in Appendix C. Exact normalization and Null-Option details are provided in Appendix D. Baseline RL algorithms are described in Appendix B.3; all controlled RL comparisons use the same backbone, training corpus, and optimization settings. The construction protocol, source datasets, and quality-control procedure for DyBench are documented in Appendix A. All main evaluations use VLMEvalKit with 32 uniformly sampled frames, as described in Section 5.1. Code, DyBench, and model checkpoints will be released.

REFERENCES

Emad Bahrami, Olga Zatsarynna, Parth Pathak, Sunando Sengupta, Juergen Gall, and Mohsen Fayyaz. Strive: Structured spatiotemporal exploration for reinforcement learning in video question answering. *arXiv preprint arXiv:2604.01824*, 2026.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025a.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025b.

Mu Cai, Reuben Tan, Jianrui Zhang, Bocheng Zou, Kai Zhang, Feng Yao, Fangrui Zhu, Jing Gu, Yiwu Zhong, Yuzhang Shang, et al. Temporalbench: Benchmarking fine-grained temporal understanding for multimodal video models. *arXiv preprint arXiv:2410.10818*, 2024.

Jang Hyun Cho, Andrea Madotto, Effrosyni Mavroudi, Triantafyllos Afouras, Tushar Nagarajan, Muhammad Maaz, Yale Song, Tengyu Ma, Shuming Hu, Suyog Jain, et al. Perceptionlm: Open-access data and models for detailed visual understanding. *arXiv preprint arXiv:2504.13180*, 2025.

Christopher Clark, Jieyu Zhang, Zixian Ma, Jae Sung Park, Mohammadreza Salehi, Rohun Tripathi, Sangho Lee, Zhongzheng Ren, Chris Dongjoo Kim, Yinuo Yang, et al. Molmo2: Open weights

and data for vision-language models with video understanding and grounding. *arXiv preprint arXiv:2601.10611*, 2026.

Daniel Cores, Michael Dorkenwald, Manuel Mucientes, Cees GM Snoek, and Yuki M Asano. Lost in time: A new temporal benchmark for videollms. *arXiv preprint arXiv:2410.07752*, 2024.

Haodong Duan, Junming Yang, Yuxuan Qiao, Xinyu Fang, Lin Chen, Yuan Liu, Xiaoyi Dong, Yuhang Zang, Pan Zhang, Jiaqi Wang, et al. Vlmevalkit: An open-source toolkit for evaluating large multi-modality models. In *Proceedings of the 32nd ACM international conference on multimedia*, pp. 11198–11201, 2024.

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. In *NeurIPS*, 2025a.

Kaituo Feng, Manyuan Zhang, Hongyu Li, Kaixuan Fan, Shuang Chen, Yilei Jiang, Dian Zheng, Peiwen Sun, Yiyuan Zhang, Haoze Sun, et al. Onethinker: All-in-one reasoning model for image and video. *arXiv preprint arXiv:2512.03043*, 2025b.

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*, pp. 24108–24118, 2025.

Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, et al. The" something something" video database for learning and evaluating visual common sense. In *Proceedings of the IEEE international conference on computer vision*, pp. 5842–5850, 2017.

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.

Wenyi Hong, Yean Cheng, Zhuoyi Yang, Weihan Wang, Lefan Wang, Xiaotao Gu, Shiyu Huang, Yuxiao Dong, and Jie Tang. Motionbench: Benchmarking and improving fine-grained video motion understanding for vision language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 8450–8460, 2025.

Lianghua Huang, Xin Zhao, and Kaiqi Huang. Got-10k: A large high-diversity benchmark for generic object tracking in the wild. *IEEE transactions on pattern analysis and machine intelligence*, 43(5): 1562–1577, 2019.

Ming Kong, Xianzhou Zeng, Luyuan Chen, Yadong Li, Bo Yan, and Qiang Zhu. Mhbench: Demystifying motion hallucination in videollms. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 4401–4409, 2025.

Benno Krojer, Mojtaba Komeili, Candace Ross, Quentin Garrido, Koustuv Sinha, Nicolas Ballas, and Mahmoud Assran. A shortcut-aware video-qa benchmark for physical understanding via minimal video pairs. *arXiv preprint arXiv:2506.09987*, 2025.

Hilde Kuehne, Ali Arslan, and Thomas Serre. The language of actions: Recovering the syntax and semantics of goal-directed human activities. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 780–787, 2014.

Jie Lei, Tamara Berg, and Mohit Bansal. Revealing single frame bias for video-and-language learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 487–507, 2023.

Baiqi Li, Kangyi Zhao, Ce Zhang, Chancharik Mitra, Jean de Dieu Nyandwi, and Gedas Bertasius. Timeblind: A spatio-temporal compositionality benchmark for video llms. *arXiv preprint arXiv:2602.00288*, 2026.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, et al. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024a.

Chaoyu Li, Eun Woo Im, and Pooyan Fazli. Vidhalluc: Evaluating temporal hallucinations in multimodal large language models for video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13723–13733, 2025a.

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*, pp. 22195–22206, 2024b.

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*, 2025b.

Yun Li, Yiming Zhang, Tao Lin, XiangRui Liu, Wenxiao Cai, Zheng Liu, and Bo Zhao. Sti-bench: Are mllms ready for precise spatial-temporal world understanding? In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5622–5632, 2025c.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023.

Shuming Liu, Mingchen Zhuge, Changsheng Zhao, Jun Chen, Lemeng Wu, Zechun Liu, Chenchen Zhu, Zhipeng Cai, Chong Zhou, Haozhe Liu, et al. Videoauto-r1: Video auto reasoning via thinking once, answering twice. *arXiv preprint arXiv:2601.05175*, 2026.

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*, pp. 8731–8772, 2024.

Jinyoung Park, Jeehye Na, Jinyoung Kim, and Hyunwoo J Kim. Deepvideo-r1: Video reinforcement fine-tuning via difficulty-aware regressive grpo. In *NeurIPS*, 2025.

Sundar Pichai, Demis Hassabis, and Koray Kavukcuoglu. A new era of intelligence with gemini 3, 2025. URL `https://blog.google/intl/en-africa/company-news/outreach-and-initiatives/a-new-era-of-intelligence-with-gemini-3/`.

Xiaoqian Shen, Yunyang Xiong, Changsheng Zhao, Lemeng Wu, Jun Chen, Chenchen Zhu, Zechun Liu, Fanyi Xiao, Balakrishnan Varadarajan, Florian Bordes, et al. Longvu: Spatiotemporal adaptive compression for long video-language understanding. *arXiv preprint arXiv:2410.17434*, 2024.

Jiapeng Shi, Junke Wang, Zuyao You, Bo He, and Zuxuan Wu. Videoloom: A video large language model for joint spatial-temporal understanding. *arXiv preprint arXiv:2601.07290*, 2026.

Yan Shu, Zheng Liu, Peitian Zhang, Minghao Qin, Junjie Zhou, Zhengyang Liang, Tiejun Huang, and Bo Zhao. Video-xl: Extra-long vision language model for hour-scale video understanding. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 26160–26169, 2025.

Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, et al. Openai gpt-5 system card. *arXiv preprint arXiv:2601.03267*, 2025.

Enxin Song, Wenhao Chai, Guanhong Wang, Yucheng Zhang, Haoyang Zhou, Feiyang Wu, Haozhe Chi, Xun Guo, Tian Ye, Yanting Zhang, et al. Moviechat: From dense token to sparse memory for long video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18221–18232, 2024.

Sebastian Stein and Stephen J McKenna. Combining embedded accelerometers with computer vision for recognizing food preparation activities. In *Proceedings of the 2013 ACM international joint conference on Pervasive and ubiquitous computing*, pp. 729–738, 2013.

Haozhe Wang, Chao Qu, Zuming Huang, Wei Chu, Fangzhen Lin, and Wenhu Chen. Vl-rethinker: Incentivizing self-reflection of vision-language models with reinforcement learning. *arXiv preprint arXiv:2504.08837*, 2025a.

Qi Wang, Yanrui Yu, Ye Yuan, Rui Mao, and Tianfei Zhou. Videorft: Incentivizing video reasoning capability in mllms via reinforced fine-tuning. In *NeurIPS*, 2025b.

Weihan Wang, Zehai He, Wenyi Hong, Yean Cheng, Xiaohan Zhang, Ji Qi, Ming Ding, Xiaotao Gu, Shiyu Huang, Bin Xu, et al. Lvbench: An extreme long video understanding benchmark. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 22958–22967, 2025c.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. Internvl3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025d.

Xiyao Wang, Zhengyuan Yang, Chao Feng, Hongjin Lu, Linjie Li, Chung-Ching Lin, Kevin Lin, Furong Huang, and Lijuan Wang. Sota with less: Mcts-guided sample selection for data-efficient visual reasoning self-improvement. *arXiv preprint arXiv:2504.07934*, 2025e.

Ziyue Wang, Sheng Jin, Zhongrong Zuo, Jiawei Wu, Han Qiu, Qi She, Hao Zhang, and Xudong Jiang. Video-ktr: Reinforcing video reasoning via key token attribution. *arXiv preprint arXiv:2601.19686*, 2026.

Zihui Xue, Mi Luo, and Kristen Grauman. Seeing the arrow of time in large multimodal models. In *NeurIPS*, 2025.

Kexin Yi, Chuang Gan, Yunzhu Li, Pushmeet Kohli, Jiajun Wu, Antonio Torralba, and Joshua B Tenenbaum. Clevrer: Collision events for video representation and reasoning. *arXiv preprint arXiv:1910.01442*, 2019.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, et al. Dapo: An open-source llm reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*, 2025.

Yuqian Yuan, Hang Zhang, Wentong Li, Zesen Cheng, Boqiang Zhang, Long Li, Xin Li, Deli Zhao, Wenqiao Zhang, Yueting Zhuang, et al. Videorefer suite: Advancing spatial-temporal object understanding with video llm. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 18970–18980, 2025.

Boqiang Zhang, Kehan Li, Zesen Cheng, Zhiqiang Hu, Yuqian Yuan, Guanzheng Chen, Sicong Leng, Yuming Jiang, Hang Zhang, Xin Li, et al. Videollama 3: Frontier multimodal foundation models for image and video understanding. *arXiv preprint arXiv:2501.13106*, 2025.

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.

Yilun Zhao, Haowei Zhang, Lujing Xie, Tongyan Hu, Guo Gan, Yitao Long, Zhiyuan Hu, Weiyuan Chen, Chuhan Li, Zhijian Xu, et al. Mmvu: Measuring expert-level multi-discipline video understanding. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 8475–8489, 2025.

Kejian Zhu, Zhuoran Jin, Hongbang Yuan, Jiachun Li, Shangqing Tu, Pengfei Cao, Yubo Chen, Kang Liu, and Jun Zhao. Mmr-v: What's left unsaid? a benchmark for multimodal deep reasoning in videos. *arXiv preprint arXiv:2506.04141*, 2025.

Orr Zohar, Xiaohan Wang, Yann Dubois, Nikhil Mehta, Tong Xiao, Philippe Hansen-Estruch, Licheng Yu, Xiaofang Wang, Felix Juefei-Xu, Ning Zhang, et al. Apollo: An exploration of video understanding in large multimodal models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 18891–18901, 2025.

A DyBench: Construction, Comparison, and Detailed Analysis

A.1 Design principle and held-out evaluation

The design of DyBench follows the recent observation that Video LLMs frequently solve dynamic questions through static shortcuts (Krojer et al., 2025; Li et al., 2026). Following the temporal-minimal-pair protocol popularized by TimeBlind (Li et al., 2026), every DyBench item is built as a pair (v, v') of videos that share the same scene and objects but differ along a single dynamic axis. The matching question q has different correct answers for v and v', so a model that exploits a single keyframe or a language prior cannot simultaneously satisfy both. Pair accuracy then reports the fraction of pairs answered correctly on *both* sides, providing a much stricter measure of dynamic understanding than per-question accuracy.

DyBench is also a strict out-of-distribution evaluation. None of its source datasets (something-something-v2 (Goyal et al., 2017), MHBench (Kong et al., 2025), GOT-10k (Huang et al., 2019), the Breakfast cooking dataset (Kuehne et al., 2014), etc.) appear in the training pool used by any of the RL methods we benchmark. Our training pool consists of Video-R1, TVBench, STI-Bench, and MMR-VBench (Appendix B). This source-level separation makes DyBench a held-out evaluation and reduces direct training–evaluation overlap as an explanation for the reported gains.

A.2 Sub-task construction

Sub-task 1: Reversible Dynamics. We collect actions whose semantics flip under temporal reversal. The action vocabulary covers humans, animals, and plants:

- *Human actions* sourced from MHBench (Kong et al., 2025), something-something-v2 (Goyal et al., 2017), and web-crawled clips: opening↔closing, picking up↔putting down, lifting↔throwing, hanging↔taking down, raising↔lowering, folding↔unfolding, pushing↔pulling, plugging in↔unplugging, zipping↔unzipping, rolling↔unrolling, tying↔untying, fastening↔unfastening, turning on↔turning off, getting in↔getting out, putting on↔taking off, inserting↔removing, stacking↔unstacking, sitting/lying/kneeling/squatting down↔standing up, inflating↔deflating, tightening↔loosening, assembling↔disassembling, leaning forward↔leaning back, bowing↔straightening up, drawing curtain open↔closed, walking forward↔backward, opening eyes↔closing eyes.
- *Animal/plant actions* sourced from web-crawled clips: standing up↔lying down, jumping up↔landing, jumping into water↔leaping out, spreading wings↔folding wings, ears up↔flattened, arching back↔relaxing, retracting into shell↔extending out, stretching neck out↔retracting, opening mouth↔closing, extending tongue↔retracting, elephant raising trunk↔lowering trunk, raising head↔lowering head, flower blooming↔closing, leaf unfolding↔folding.

For each source clip we generate v (forward play) and v' (reverse play). The question asks "Which of the following best describes the action in the video?" with options drawn from the forward action, the reversed action, and a distractor. The answer flips between v and v'.

Sub-task 2: Moving Direction. We use object-tracking videos from GOT-10k (Huang et al., 2019) and motion-direction questions from CLEVRER (Yi et al., 2019). For each source clip we generate v (original) and v' (horizontally flipped, optionally also temporally reversed). The question asks "In what direction is the [object] moving?" with the four standard direction options. For tracking videos with non-axial motion (e.g., diagonals), the option set is augmented with the four diagonals.

Sub-task 3: Event Sequence. We assemble pairs of two-event composite videos from four sources, including Breakfast (Kuehne et al., 2014), 50Salads (Stein & McKenna, 2013), VIDHALLUC (Li et al., 2025a) and TempCompass (Liu et al., 2024). For Breakfast and 50Salads, we follow the original action annotations and sample two non-adjacent action segments per video (each $\leq$ 30s). The two segments are concatenated to form v and concatenated in the reverse order to form v'. For VIDHALLUC, we use cached DINO features to find the splice point, keeping only those clips with exactly one detected splice. Each clip is then re-spliced in reverse order to form v'.

Quality control. Every pair undergoes manual verification of (i) content consistency between v and v', (ii) counterfactual validity, so that no unintended task-relevant cue changes beyond the designed intervention, and (iii) unambiguous correctness of the answer for both videos. Pairs that fail any check are discarded.

A.3 Comparison with existing video benchmarks

Table 3 compares DyBench with widely used video benchmarks along four dimensions: scale, video length, domain focus, and whether the benchmark uses adversarial pair construction. DyBench is most closely related to TimeBlind (Li et al., 2026) in spirit, since both use minimal pair construction and paired evaluation, but DyBench focuses specifically on three motion-aware sub-tasks (reversible dynamics, moving direction, and event sequence) and reports pair accuracy as the primary metric. In addition, the DyBench dataset is much larger.

Table 3: DyBench vs. existing video benchmarks. "Adversarial" means each item is paired with a counter-example designed to flip the answer.

Benchmark	#Videos / #QAs	Length	Domain	Adversarial
MVBench (Li et al., 2024b)	4,000 / 4,000	short	general	No
VideoMME (Fu et al., 2025)	900 / 2,700	short, long	general	No
LVBench (Wang et al., 2025c)	103 / 1,549	long	general	No
TempCompass (Liu et al., 2024)	410 / 1,580	short	dynamic	Yes
TemporalBench (Cai et al., 2024)	2,000 / 10,000	short, long	dynamic	No
MotionBench (Hong et al., 2025)	5,385 / 8,052	short	human action	No
MHBench (Kong et al., 2025)	1,200 / 3,600	short	human action	Yes
TimeBlind (Li et al., 2026)	1,200 / 2,400	short	dynamic	Yes
DyBench (Ours)	**3,014 / 3,014**	short	dynamic	**Yes**

A.4 Shortcut-isolation analysis

A central design goal of DyBench is to make static shortcuts ineffective under paired evaluation. We verify this by re-evaluating models under three handicapped input settings, in the spirit of the shortcut analyses of TimeBlind (Li et al., 2026) and MVP (Krojer et al., 2025).

- **Single Frame.** The model receives only a single, randomly sampled frame from each video.
- **Shuffled Frames.** The model receives all 32 frames but in a uniformly random order, destroying temporal structure while preserving every static cue.
- **Text Only.** The model receives the question and options without any visual input.

A benchmark that is genuinely diagnostic of spatiotemporal understanding should drive accuracy to near-chance under all three settings. Table 4 reports the results. Because DyBench mixes 2-, 3-, and 4-way questions, the random-chance overall is 35.1% for Acc and 13.3% for P-Acc; per sub-task, the chance values are $33.3\%/11.1\%$ for Reversible Dynamics (3-way), $25.0\%/6.3\%$ for Moving Direction (4-way), and $50.0\%/25.0\%$ for Event Sequence (binary).

Table 4: **Shortcut-isolation analysis on DyBench.** "Full Video" is the standard 32-frame setting reproduced from Table 1; the other three columns are handicapped inputs that destroy temporal information.

Model	Full Video		Single Frame		Shuffled Frames		Text Only	
	Acc	P-Acc	Acc	P-Acc	Acc	P-Acc	Acc	P-Acc
GPT-5.1	63.7	44.9	39.9	5.1	30.5	3.0	32.8	2.6
Gemini-3.1-Pro	82.2	71.7	43.3	6.8	41.6	7.7	42.8	6.1
Qwen3-VL-8B	68.2	50.4	35.1	7.2	42.3	10.2	27.7	0.4
+ **CRPO (Ours)**	72.5	58.1	37.9	7.4	44.3	13.2	36.8	1.1
Random chance	35.1	13.3	35.1	13.3	35.1	13.3	35.1	13.3

We find that **under every handicapped input, all models collapse to or below chance on the strict pair metric**. CRPO drops from 58.1 P-Acc on the full video to 7.4 (Single Frame), 13.2 (Shuffled Frames), and 1.1 (Text Only), well below the 13.3 chance level for two of the three settings. The same pattern holds for GPT-5.1 (P-Acc collapses from 44.9 to $\leq$ 5.1), Gemini-3.1-Pro and the Qwen3-VL-8B baseline. This validates DyBench as a benchmark that genuinely requires temporal information, since shuffling frames, removing all but one frame, or removing the video entirely all destroy the signal needed to solve it. The Text-Only P-Acc values are small but not exactly zero because some construction pipelines preserve the same semantic answer choices while changing their presentation order across the two sides of a pair, so the textual prompts are not always byte-identical.

A.5 Per-sub-task breakdown

Table 5 decomposes DyBench accuracy into the three sub-tasks introduced in Sec. 4.

Table 5: **Per-sub-task DyBench results.** Both per-question accuracy (Acc) and pair accuracy (P-Acc) are reported on each sub-task.

Model	Reversible Dynamics		Moving Direction		Event Sequence		Overall	
	Acc	P-Acc	Acc	P-Acc	Acc	P-Acc	Acc	P-Acc
GPT-5.1	80.7	66.0	31.2	9.6	78.5	57.0	63.7	44.9
Gemini-3.1-Pro	96.1	92.3	57.7	36.8	91.9	84.3	82.2	71.7
Qwen3-VL-4B (base)	78.9	64.3	40.1	18.0	75.4	51.1	65.1	45.4
+ GRPO	82.0	67.3	39.0	19.0	77.9	56.0	66.6	48.2
+ T-GRPO	82.0	68.3	**44.7**	**21.8**	78.3	57.0	68.6	49.8
+ ArrowRL	80.8	66.4	40.1	17.2	78.5	57.0	66.7	47.5
+ **CRPO (Ours)**	**86.8**	**77.7**	41.8	21.0	**81.1**	**62.7**	**70.3**	**54.8**
Qwen3-VL-8B (base)	83.3	70.8	40.8	18.2	79.5	59.7	68.2	50.4
+ GRPO	84.0	71.8	41.4	17.6	82.4	66.3	69.4	52.4
+ T-GRPO	85.5	74.3	**43.5**	**20.0**	83.3	67.6	70.9	54.5
+ ArrowRL	84.2	71.2	42.0	19.4	82.1	64.6	69.6	52.2
+ **CRPO (Ours)**	**89.8**	**81.5**	41.7	19.8	**85.0**	**70.5**	**72.5**	**58.1**
Random chance	33.3	11.1	25.0	6.3	50.0	25.0	35.1	13.3

Three observations stand out. **(i)** CRPO yields its largest gains on Reversible Dynamics and Event Sequence, the two sub-tasks that most directly require temporal reasoning. In particular, Event Sequence is constructed through segment reordering, which is never used as a CRPO training transformation; CRPO reaches 62.7 P-Acc at 4B and 70.5 at 8B, compared with 57.0 and 67.6 for T-GRPO. **(ii)** Moving Direction remains challenging for all open-source models. T-GRPO is slightly higher than CRPO on this sub-task (21.8 vs. 21.0 P-Acc at 4B and 20.0 vs. 19.8 at 8B), showing that CRPO's gains are not uniform across every DyBench category. **(iii)** The much larger separation on P-Acc than on per-question Acc illustrates why paired evaluation is useful for diagnosing sensitivity to the underlying video dynamics.

B Training Data and Implementation Details

B.1 Training data and reward functions

Data sources. The training corpus combines text, image, and video QA, summarized in Table 6. The dual-branch counterfactual framework of CRPO is applied only to video samples, where spatial and temporal perturbations are meaningful. Text and image samples are trained with standard single-branch GRPO. Following standard practice, samples for which the base model is either always correct or always wrong across 8 base-model rollouts are filtered out, since they carry no signal under group-relative advantages. The video subset contains 21,165 samples after filtering, whose distribution across Task Router categories is reported in Appendix C.

Reward functions. We use two standard reward terms shared by all RL methods, namely a QA-style *accuracy* reward and a *format* reward. The accuracy reward is binary, with math problems

Table 6: Training data composition.

Modality	Sources	Approx. size
Text	DAPO-Math (Yu et al., 2025)	6.4K
Image	VIRL (Wang et al., 2025a), ThinkLite-VL-Hard (Wang et al., 2025e)	27.5K
Video	Video-R1 (Feng et al., 2025a), TVBench (Cores et al., 2024), STI-Bench (Li et al., 2025c), MMR-VBench (Zhu et al., 2025)	21.2K

verified by `math-verify` and other QA checked by case-folded, whitespace-stripped string match. The format reward is binary and enforces a strict regex of one ⟨`think`⟩ block followed by a single `\boxed{}`. CRPO additionally introduces R_{CRR} and R_{behave} as defined in Section 3.2.

Training prompt. All RL methods (CRPO and the baselines) use the following RL-with-Thinking system prompt during training.

Training System Prompt

You are a helpful assistant. **FIRST**, think through the reasoning process as an internal monologue, and **THEN** provide the final answer. The reasoning process MUST be enclosed within `<think> </think>` tags, and the final answer MUST be wrapped in `\boxed{}`.

For multiple-choice questions, the option list is appended to the user prompt together with the appended "None of the above" (Appendix D.1).

B.2 Hyperparameters

Table 7 lists the key hyperparameters used by all RL methods (GRPO, T-GRPO, ArrowRL, and CRPO). Hyperparameters that differ between methods are limited to the CRPO-specific block at the bottom of the table.

Table 7: Key training hyperparameters. The first block is shared by all RL baselines; the second block is CRPO-specific.

Hyperparameter	Value
Shared (all RL methods)	
Backbone	Qwen3-VL-4B / 8B-Instruct (Bai et al., 2025a)
Tunable modules	LLM + multimodal projector (vision encoder frozen)
Optimizer	AdamW (fused), β_1=0.9, β_2=0.999, weight decay 0.01
Learning rate / schedule	1×10^{-6}, constant with warmup
KL coefficient β	0.01
Max gradient norm	1.0
Batch size (per device)	4
Generation steps per update	8
Rollouts per group G	8
Max prompt / completion length	8192 / 2048
Sampled frames per video	32 (uniform)
Pixel budget per video	$\leq 2048\times32\times32$
Pixel budget per frame	$[16,\ 768]\times32\times32$
Epochs	1
Precision	BF16, ZeRO-3, gradient checkpointing
Rollout backend	vLLM (colocate, TP = 8, GPU mem util. 0.45)
Hardware	32 NVIDIA H20 GPUs (4 nodes × 8)
CRPO-specific	
Cross-branch CRR weight λ_d=λ_s	0.3
Counterfactual branch weight w_{aug}	0.5
Counterfactual transformations	horizontal flip; temporal reversal

Evaluation toolkit. All evaluations are run with the public VLMEvalKit (Duan et al., 2024) toolkit[1] with its default configuration of 32 uniformly sampled frames per video for video benchmarks. We do not change any benchmark-side prompts or option ordering, and we use greedy decoding.

B.3 Baseline RL algorithms

For a fair comparison, all RL post-training baselines (GRPO, T-GRPO, ArrowRL) are reproduced under the same backbone, training data, and shared hyperparameters as CRPO. They differ only in the RL algorithm. We reimplement each baseline strictly following its original paper.

GRPO (Guo et al., 2025). The vanilla baseline. For each prompt, the policy generates G=8 rollouts, and each rollout receives the standard correctness reward $R_{\text{correct}} = \mathbb{I}[o = o^*]$ plus the format reward R_{format}. Advantages are computed by subtracting the group mean reward and normalizing by the group standard deviation, and the policy is updated with the clipped GRPO surrogate objective.

T-GRPO (Feng et al., 2025a). Following Video-R1, for each prompt we additionally generate a second group of G=8 rollouts on a frame-shuffled version of the input video. Let p and $\tilde{p}$ denote the fraction of correct rollouts in the ordered and shuffled groups, respectively. We add a temporal reward $r_t = \alpha \cdot \mathbb{I}[p \geq \tilde{p}]$ with $\alpha = 0.3$ to each correct rollout in the ordered group. Following the original paper, the shuffled group is used only to compute $\tilde{p}$ and does *not* contribute to the policy gradient, so the optimization is effectively single-branch. The temporal reward is applied only to video samples.

ArrowRL (Xue et al., 2025). Following the original paper, for each prompt we generate one rollout group on the original video and a single reference response $\tilde{o}$ on the temporally reversed video. The reward for each original-video rollout o_i combines a fidelity term and a reverse term, $r_i = r_i^{\text{fid}} + \alpha_i \cdot r_i^{\text{rev}}$, where $r_i^{\text{fid}} = \mathbb{I}[o_i = o^*]$ for multiple-choice questions, $r_i^{\text{rev}} = 1 - \mathbb{I}[o_i = \tilde{o}]$ penalizes rollouts that mirror the reverse-conditioned response, and α_i is a dynamic weight set to $\alpha = 0.25$ for AoT-sensitive samples ($\tilde{o} \neq o^*$) and 0 otherwise ($\tilde{o} = o^*$). The reversed video is used only to compute $\tilde{o}$ for reward shaping and does not contribute policy gradients.

C Task Router

Classification model. The Task Router is run *offline* once on the entire training set with DeepSeek-R1, a text-only reasoning model. The classification is text-only and therefore does not require access to the video itself.

Prompt template. The full prompt template is reproduced below. It is the verbose implementation of the two imagination questions shown in Figure 2 (right panel), with detailed criteria for what counts as a YES versus a NO under each test. Since the Task Router classifies only video samples, the prompt is designed around video-specific transformations (horizontal flip and temporal reversal).

Task Router Prompt (used with DeepSeek-R1)

Task. Classify this VQA item into [`Temporal`, `Spatial`, `Spatiotemporal`, `Static`] based on two transformation tests.

1. Horizontal Flip Test (Spatial Sensitivity). If the video is mirrored horizontally (Left ↔ Right):

- **YES (Change)**: queries about left/right, clockwise/counter-clockwise, or horizontal orientation.
- **NO (No Change)**: queries about verticality (top/bottom, high/low, tall/short), color, identity, size, or count.

2. Time-Reversal Test (Temporal Sensitivity). If the video is played in REVERSE (frames played backwards, time flows backward):

- **YES (Change)**: directional actions where forward ≠ backward (opening→closing, picking up→putting down, entering→exiting); questions about FIRST/LAST or BEFORE/AFTER; questions about cause/result of a sequence.
- **NO (No Change)**: counting, identity/color/material/shape/size, persistent states or attributes, motion speed that are symmetric under reversal.

[1] `https://github.com/open-compass/VLMEvalKit`

Decision Matrix.

- `Spatial`: Flip **YES** AND Time-Reversal **NO**.
- `Temporal`: Flip **NO** AND Time-Reversal **YES**.
- `Spatiotemporal`: **both YES**.
- `Static`: **both NO**.

Data. Q: `{problem}` Options: `{options}` A: `{solution}`

Instructions. First, briefly reason through the two tests. Then, on the very last line, output ONLY one word from {`Temporal`, `Spatial`, `Spatiotemporal`, `Static`}.

Statistics on the training set. Table 8 reports the per-source distribution. The video subset is dominated by `Temporal` questions (73.6%), reflecting that most existing video QA data is action- or event-centric. Pure `Spatial` questions are rare (2.3%), and `Spatiotemporal` questions, which require both axes, account for 4.1%. The `Static` category (20.0%) covers static questions about presence, count, color, etc.

Table 8: Task Router classification statistics on the video training set (21,165 samples).

Dataset	Temporal	Spatial	Spatiotemp.	Static	Total
Video-R1	14,359 (80.8%)	173 (1.0%)	310 (1.7%)	2,936 (16.5%)	17,778
TVBench	979 (70.9%)	2 (0.1%)	238 (17.2%)	161 (11.7%)	1,380
STI-Bench	81 (5.3%)	313 (20.5%)	319 (20.8%)	817 (53.4%)	1,530
MMR-VBench	163 (34.2%)	3 (0.6%)	1 (0.2%)	310 (65.0%)	477
Total	**15,582 (73.6%)**	**491 (2.3%)**	**868 (4.1%)**	**4,224 (20.0%)**	**21,165**

Validation by manual relabeling. To verify the router's quality, we randomly sample 200 video QA examples and manually relabel them by inspecting the video, question, and options. The router agrees with the human labels on 188 of 200 cases (**94%**). Eight of the 12 disagreements are borderline `Temporal`/`Spatiotemporal` cases, which still select temporal reversal as a valid transformation; the remaining four involve `Static` or `Spatial` mismatches.

Robustness to routing errors. The manual audit above measures the Task Router's prediction quality, but does not directly reveal how sensitive CRPO is to routing errors. We therefore conduct a controlled noise experiment on Qwen3-VL-4B, randomly corrupting the router assignments of 10% or 20% of the video training samples while keeping all other training settings unchanged. Table 9 reports the results. A 10% corruption rate causes only small changes, reducing DyBench P-Acc from 54.8 to 54.2 and TimeBlind I-Acc from 31.7 to 31.3. Performance degrades more clearly at 20% corruption, but remains above the Qwen3-VL-4B base model on both paired benchmarks. Together with the 94% agreement in the manual audit, these results indicate that CRPO does not require a perfectly accurate router.

Table 9: **Robustness to Task Router errors on Qwen3-VL-4B.** We randomly corrupt the router assignments for a fraction of video training samples and retrain CRPO with all other settings unchanged.

Router noise	DyBench		TimeBlind	
	Acc	P-Acc	Acc	I-Acc
0%	70.3	54.8	69.8	31.7
10%	70.0	54.2	69.6	31.3
20%	68.6	51.4	67.9	29.4

Example router outputs. Figure 5 shows representative router outputs across the four categories, including the question and ground-truth answer.

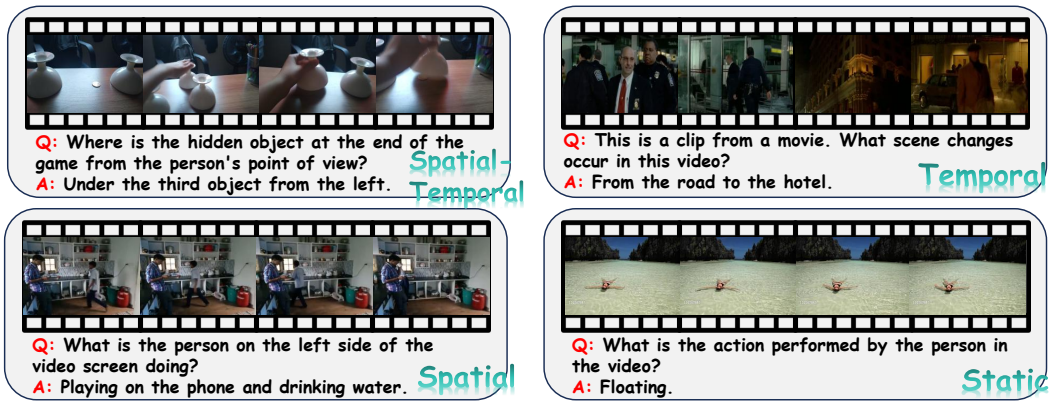

Figure 5: **Task Router output examples** after applying the prompt above to randomly sampled videos. The router outputs a category label together with a short text-only reasoning chain.

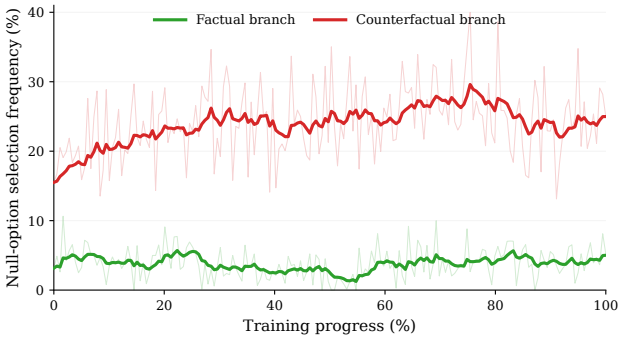

Figure 6: **Null-Option selection frequency during Qwen3-VL-4B training.** Thin curves show per-batch frequencies and thick curves show smoothed trends. Null-Option usage remains low on factual rollouts, while counterfactual rollouts fluctuate at a substantially higher level and stabilize around 25% rather than increasing continuously.

D Additional CRPO Implementation Details

D.1 Null Option

For multiple-choice training examples, a flip or temporal reversal can move the correct counterfactual answer outside the original answer choices. We therefore append a *None of the above* option $o^{\mathcal{N}}$ during RL training. On the factual branch, the original ground-truth answer remains present, so the Null Option is always incorrect. On a dynamic counterfactual branch, it allows the policy to express that none of the original choices remains valid after the intervention; on Static questions it remains incorrect because the answer should be preserved. The Null Option is used only during training and is not added to benchmark evaluations. Its contribution is quantified by the "w/o Null Option" row in Table 2.

Selection dynamics. A potential concern is that the policy may exploit the Null Option by selecting it indiscriminately on counterfactual inputs. Figure 6 tracks its per-batch selection frequency during Qwen3-VL-4B training. On the factual branch, the frequency fluctuates across batches but remains low on average, with the smoothed trend staying around a few percent throughout training. In contrast, the counterfactual branch exhibits substantially higher usage: per-batch frequencies vary roughly between 12% and 40%, while the smoothed trend stabilizes around 25% after the early stage of training. Importantly, the frequency does not continue to increase with training, but instead fluctuates around a stable level. This branch-specific pattern is consistent with the intended role of the Null Option: counterfactual transformations can invalidate the original answer choices, whereas

factual inputs retain an explicit valid target. Together with factual correctness supervision and the invariance constraint on Static examples, this behavior argues against a degenerate always-Null strategy. The ablation in Table 2 further shows that removing the Null Option substantially reduces paired performance.

D.2 Exact advantage normalization

Let R_i^{F} and R_j^{CF} be the factual and counterfactual rewards from Eqs. 1–2. CRPO first centers the two rollout groups separately,

$$\delta_i^{\mathrm{F}} = R_i^{\mathrm{F}} - \frac{1}{G}\sum_{k=1}^{G} R_k^{\mathrm{F}}, \tag{5}$$

$$\delta_j^{\mathrm{CF}} = R_j^{\mathrm{CF}} - \frac{1}{G}\sum_{k=1}^{G} R_k^{\mathrm{CF}}. \tag{6}$$

It then computes a shared scale over the $2G$ centered rewards,

$$\sigma_{\mathrm{shared}} = \sqrt{\frac{1}{2G}\left(\sum_{i=1}^{G}(\delta_i^{\mathrm{F}})^2 + \sum_{j=1}^{G}(\delta_j^{\mathrm{CF}})^2\right) + \epsilon}, \tag{7}$$

and uses

$$\widehat{A}_i^{\mathrm{F}} = \frac{\delta_i^{\mathrm{F}}}{\sigma_{\mathrm{shared}}}, \qquad \widehat{A}_j^{\mathrm{CF}} = \frac{\delta_j^{\mathrm{CF}}}{\sigma_{\mathrm{shared}}}. \tag{8}$$

Thus, each branch preserves its own within-group ranking while both branches are placed on the same optimization scale. We apply the standard GRPO token-level KL regularizer to the reference policy on both branches, using the same KL coefficient $\beta = 0.01$ as in the controlled baselines (Table 7).

D.3 Auxiliary-reward cancellation under group-wise normalization

The cancellation discussed in Section 3.4 arises in the common case where a perturbation-derived bonus acts as the same multiplicative factor on all correct rollouts in a group. Ignoring the format term for clarity, let the binary correctness reward be $c_i \in \{0, 1\}$ and let a group-level signal produce $R_i = (1 + C)c_i$. If $\bar{c}$ and σ_c are the mean and standard deviation of c_i, then the normalized advantage is

$$\widehat{A}_i = \frac{(1 + C)(c_i - \bar{c})}{(1 + C)\sigma_c + \epsilon} \approx \frac{c_i - \bar{c}}{\sigma_c}, \tag{9}$$

where the approximation becomes exact as the numerical stabilizer becomes negligible. The auxiliary factor therefore contributes little or no new relative advantage in this case. This analysis does not imply that every perturbation-based reward is always cancelled; it identifies a structural failure mode when the auxiliary signal is shared across the rollout group.

CRPO receives a qualitatively different signal because the counterfactual branch generates independent rollouts with per-rollout behavioral rewards b_j. Some counterfactual rollouts satisfy the expected response and others do not, producing within-group variation that survives normalization and directly contributes policy gradients. In addition, CRPO's shared cross-branch standard deviation prevents each branch from independently rescaling away its cross-branch reward factor. These two properties explain why counterfactual information enters CRPO through the optimized rollouts themselves rather than only through a group-level reward modifier.

E Additional Training Analysis

E.1 The four CRR reward components

Figure 7 visualizes the four reward terms introduced by CRPO. The original branch combines a correctness reward R_{correct} and a cross-branch CRR reward $R_{\text{CRR}}^{\text{orig}}$ that scores the counterfactual

rollouts. The counterfactual branch combines a behavioral reward R_{behave} and a cross-branch CRR reward $R_{\text{CRR}}^{\text{aug}}$ that scores the original rollouts. Both branches additionally use the inherited format reward.

A natural observation from Figure 7 is that the two cross-branch curves $R_{\text{CRR}}^{\text{orig}}$ and $R_{\text{CRR}}^{\text{aug}}$ track each other almost perfectly throughout training. This is not a plotting artifact but a direct consequence of the symmetric mutual-reward design. Let p denote the fraction of correct rollouts in the original branch and q the fraction of behavior-matching rollouts in the counterfactual branch. By construction, $R_{\text{CRR}}^{\text{orig}}$ activates only on the pG correct original rollouts and each of them measures how often the counterfactual rollouts behave as expected, giving a batch-level expectation of $\lambda \cdot p \cdot q$. The counterfactual side is analogous: $R_{\text{CRR}}^{\text{aug}}$ activates only on the qG behavior-matching counterfactual rollouts and each measures the original branch's correctness rate p, again yielding $\lambda \cdot p \cdot q$. The two CRR streams therefore receive signals of equal magnitude and neither branch's reward dominates the other.

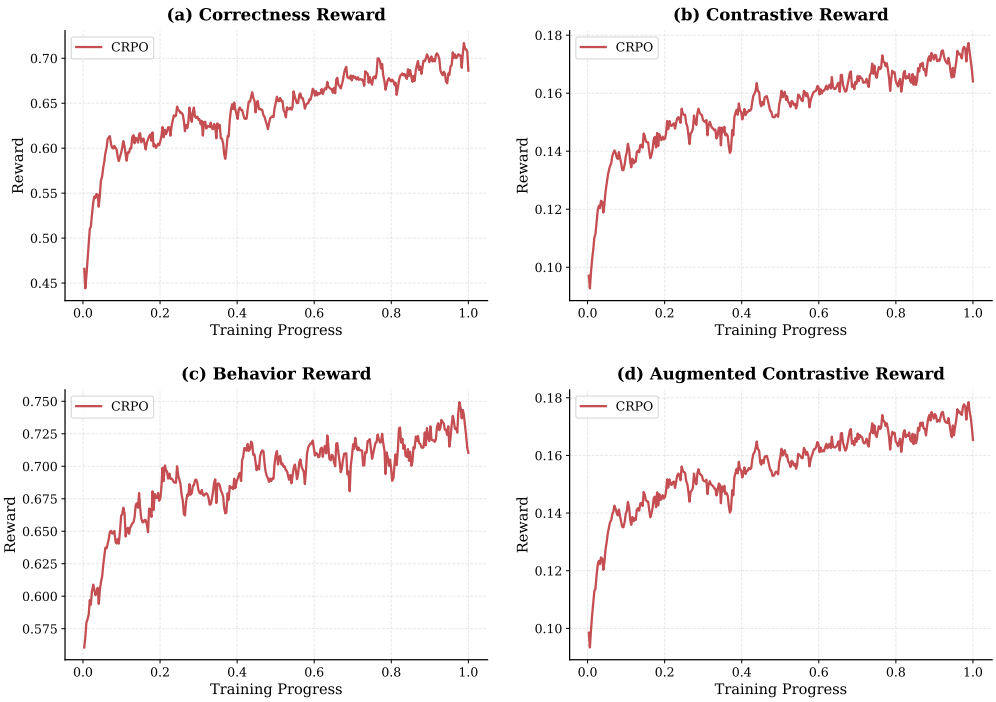

Figure 7: **The four reward components introduced by CRPO.**

E.2 Training dynamics on Qwen3-VL-8B

Figure 8 plots the same three training-curve diagnostics as Figure 4 in the main paper for Qwen3-VL-8B. The trends mirror the 4B results: CRPO's factual correctness rises more gradually before converging to a comparable level, the fraction of zero-advantage groups remains lower, and the relational reward grows during training while the perturbation-derived rewards of T-GRPO and ArrowRL remain relatively flat. The similarity across the two backbone sizes indicates that the observed optimization behavior is not specific to the 4B model.

E.3 Sensitivity to reward weights

To probe how robust CRPO is to its two reward coefficients, we run two one-dimensional sweeps on Qwen3-VL-4B and report two complementary metrics, the strict paired metric **DyBench P-Acc** (which most directly reflects spatiotemporal sensitivity) and the standard accuracy **VideoMME** (which reflects general video understanding).

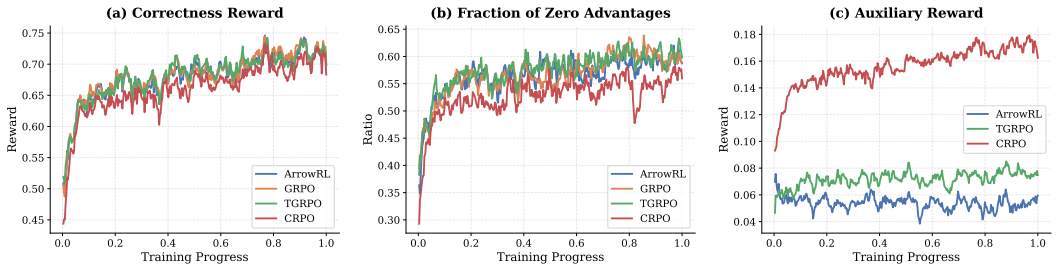

Figure 8: **Training dynamics of CRPO vs. RL baselines on Qwen3-VL-8B.** Same layout as Figure 4 (4B). The trends are consistent across model scales.

- **Sweep A.** Vary the cross-branch CRR weight $\lambda{=}\lambda_d{=}\lambda_s$ over $\{0.0, 0.1, 0.3, 0.5\}$ with $w_{\text{aug}}{=}0.5$ fixed. The point $\lambda{=}0$ corresponds to a dual-branch GRPO without any CRR signal, isolating the contribution of the relational reward.
- **Sweep B.** Vary the counterfactual-branch weight w_{aug} over $\{0.0, 0.3, 0.5, 0.7\}$ with $\lambda{=}0.3$ fixed. The point $w_{\text{aug}}{=}0$ disables direct policy updates from the counterfactual branch while retaining the counterfactual rollouts used to compute the factual-branch relational reward.

Figure 9 plots the results.

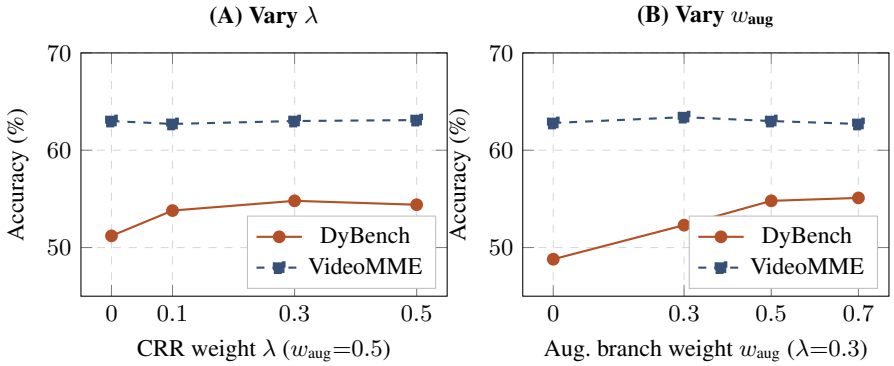

Figure 9: **Sensitivity of CRPO to its two reward coefficients** on Qwen3-VL-4B. Solid orange: DyBench P-Acc (spatiotemporal-sensitive); dashed navy: VideoMME (general). *Left:* varying the cross-branch CRR weight λ with $w_{\text{aug}}{=}0.5$ fixed. *Right:* varying the counterfactual-branch weight w_{aug} with $\lambda{=}0.3$ fixed.

Takeaways. Setting $\lambda{=}0$ lowers DyBench P-Acc from 54.8 to 51.2, showing that the cross-branch relational term contributes beyond the local branch rewards. Setting $w_{\text{aug}}{=}0$ produces a larger drop to 48.8, consistent with the main ablation showing that direct policy updates from counterfactual rollouts are especially important. Across the non-zero settings, DyBench P-Acc changes modestly, while VideoMME remains within a narrow range, indicating that CRPO does not depend on a finely tuned coefficient choice.

F Additional Qualitative Examples

Figures 10–15 show additional qualitative comparisons between Qwen3-VL (baseline) and CRPO. The first four examples (Figures 10–13) are drawn from **DyBench**, while the last two (Figures 14–15) are drawn from **TimeBlind** (Li et al., 2026). Across these examples, CRPO answers both sides of each pair correctly, while the baseline either repeats the same answer across the pair or predicts an incorrect direction or event order.

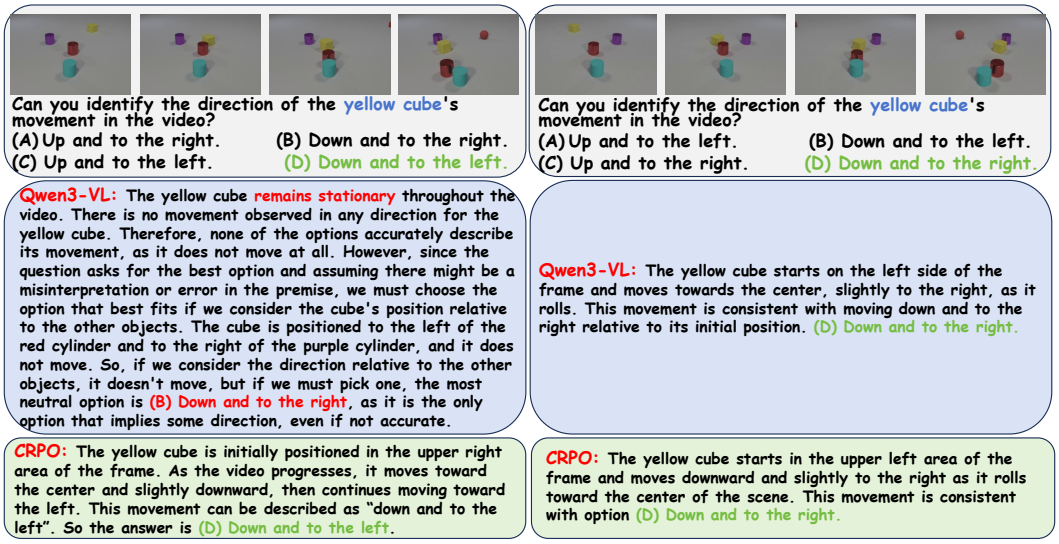

Figure 10: Qualitative example from DyBench (*moving direction*): identifying which way the yellow cube moves on opposite paired videos.

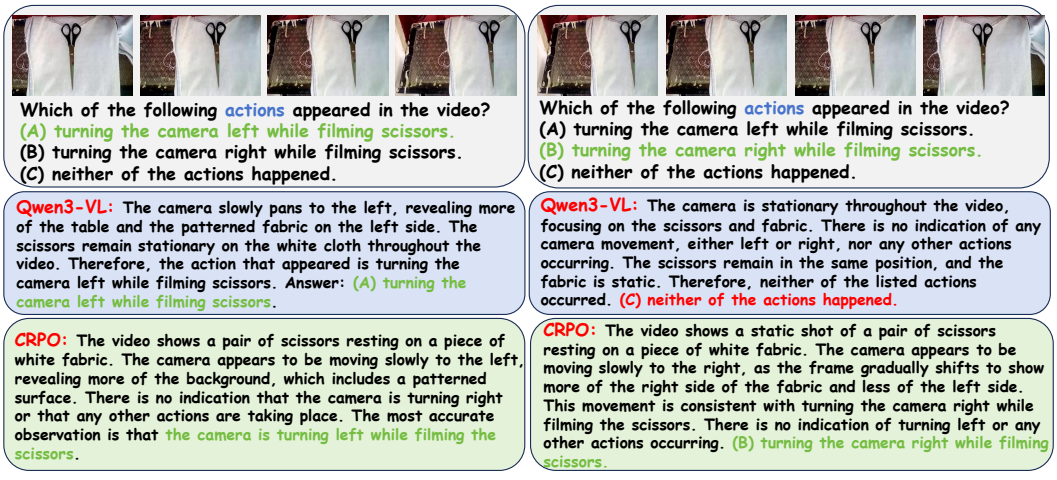

Figure 11: Qualitative example from DyBench (*moving direction*): identifying whether the camera pans left or right while filming the scissors.

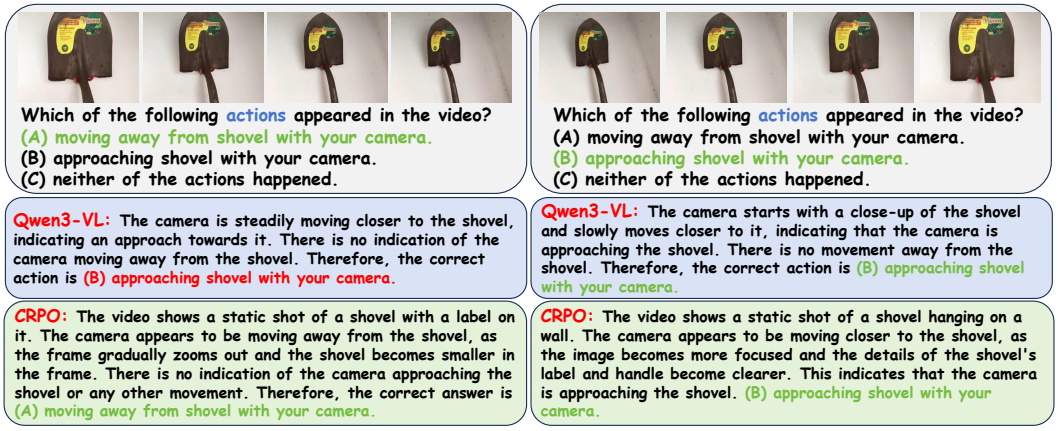

Figure 12: Qualitative example from DyBench (*moving direction*): identifying whether the camera is approaching or moving away from the shovel.

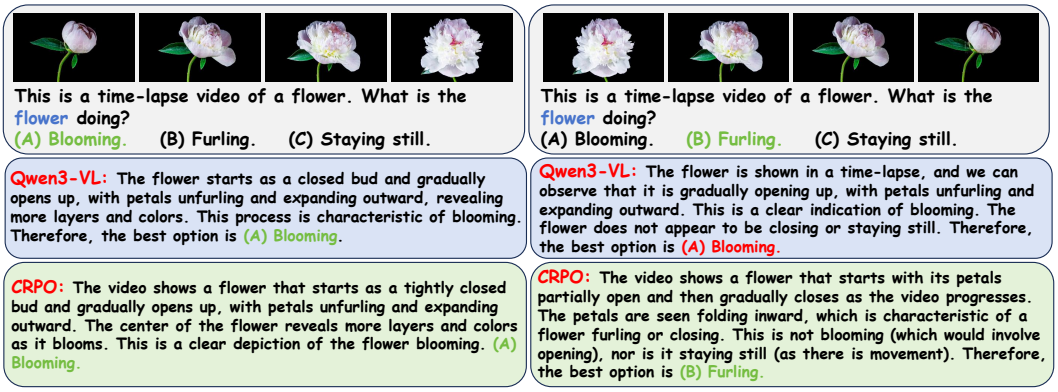

Figure 13: Qualitative example from DyBench (*reversible dynamics*): identifying whether a time-lapse of a flower shows blooming or furling.

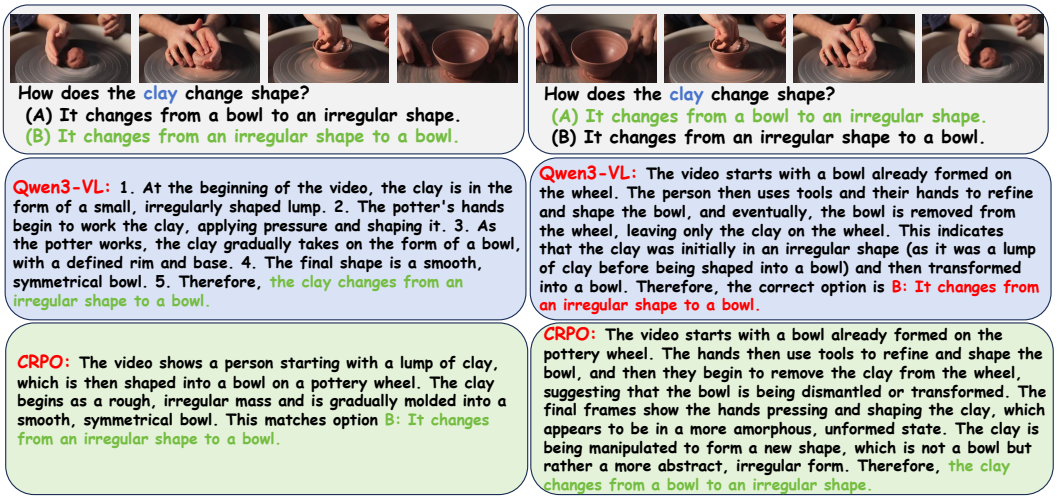

Figure 14: Qualitative example from TimeBlind: identifying whether the clay changes from a lump to a bowl, or from a bowl to a lump.

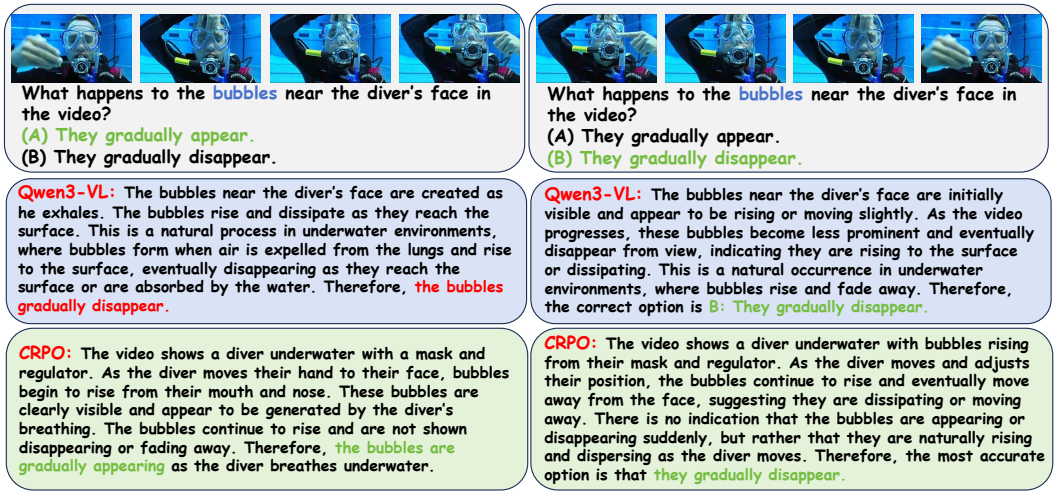

Figure 15: Qualitative example from TimeBlind: identifying whether the bubbles near the diver's face gradually appear or gradually disappear.

G Limitations and Broader Impact

Limitations. CRPO currently uses horizontal flipping and temporal reversal as broadly applicable counterfactual transformations. Extending the framework to richer semantic interventions, such as object substitution or state editing, would require reliable video editing or curated counterfactual data. The Task Router also introduces a one-time offline preprocessing step.

Broader impact. CRPO is intended to make Video LLM predictions more responsive to the spatiotemporal evidence in the input, which may improve reliability in applications that depend on motion direction or event order. It does not eliminate model errors or provide guarantees for safety-critical use, and the broader dual-use considerations remain those of Video LLMs and multimodal foundation models more generally.