DELTARUBRIC: GENERATIVE MULTIMODAL REWARD MODELING VIA JOINT PLANNING AND VERIFICATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Aligning Multimodal Large Language Models (MLLMs) requires reliable reward models, yet existing single-step evaluators can suffer from lazy judging, exploiting language priors over fine-grained visual verification. While rubric-based evaluation mitigates these biases in text-only settings, extending it to multimodal tasks is bottlenecked by the complexity of visual reasoning. The critical differences between responses often depend on instance-specific visual details. Robust evaluation requires dynamically synthesizing rubrics that isolate spatial and factual discrepancies. To address this, we introduce **DeltaRubric**, an approach that reformulates multimodal preference evaluation as a plan-and-execute process within a single MLLM. DeltaRubric operates in two steps: acting first as a *Disagreement Planner*, the model generates a neutral, instance-specific verification checklist. Transitioning into a *Checklist Verifier*, it executes these self-generated checks against the image and question to produce the final grounded judgment. We formulate DeltaRubric as a multi-role reinforcement learning problem, jointly optimizing planning and verification capabilities. Validated on Qwen3-VL 4B and 8B Instruct models, DeltaRubric achieves solid empirical gains, improving base-model overall accuracy on VL-RewardBench by **+22.6** (4B) and **+18.8** (8B) points and consistently surpassing no-rubric baselines. Beyond accuracy, DeltaRubric outperforms compute-matched inference scaling (verbose CoT and best-of-N), preference learning (DPO), and automated prompt optimization while generating fewer tokens, generalizes to a different backbone InternVL3 and to scalar rewards, and as a frozen reward model improves downstream policy training. The results demonstrate that decomposing evaluation into structured, verifiable steps leads to more reliable and generalizable multimodal reward modeling.

1 INTRODUCTION

Reinforcement Learning from Human Feedback (RLHF) (Ouyang et al., 2022; Bai et al., 2022) has become the de facto standard for aligning Large Language Models (LLMs) with human intentions and values. At its core lies the Reward Model (RM) (Zhong et al., 2025; Lambert, 2025; Rafailov et al., 2023), which serves as a proxy for human preference by scoring or comparing candidate responses and guiding policy optimization. For easy-to-verify tasks such as mathematical reasoning and coding, alignment can often be achieved with rule-based verifiers Trung et al. (2024); Lambert et al. (2024); Guo et al. (2025); Zheng et al. (2025a); Dai et al. (2025); Liu et al. (2025b;a); Li et al. (2025b). In contrast, open-ended and hard-to-verify tasks rely on learned reward models, which demand extensive human annotations to approximate nuanced preferences.

Recent advances have sought to move beyond scalar reward signals. In text-only settings, reward modeling has evolved from predicting scalar scores (Ouyang et al., 2022; Bai et al., 2022) to LLM-as-a-judge frameworks (Saunders et al., 2022; Zheng et al., 2023; Kim et al., 2023), which generate both preference judgments and Chain-of-Thought (CoT) rationales (Wei et al., 2022). To better capture the multidimensional nature of response quality in open-ended tasks, there is a growing trend toward adopting rubric-based evaluation (Gunjal et al., 2025; Liu et al., 2025c; Huang et al., 2025; Shao et al., 2025; Xu et al., 2026), including the most recent DeepSeek-V4 (DeepSeek-AI, 2026), demonstrating that decomposing a complex judgment into a set of criteria effectively improves evaluator reliability and generalization.

The transition toward Multimodal Large Language Models (MLLMs) introduces new alignment challenges (Sun et al., 2024; Yu et al., 2024; Chen et al., 2024a). Extending RLHF to the visual domain requires multimodal reward models capable of assessing the consistency between textual

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.

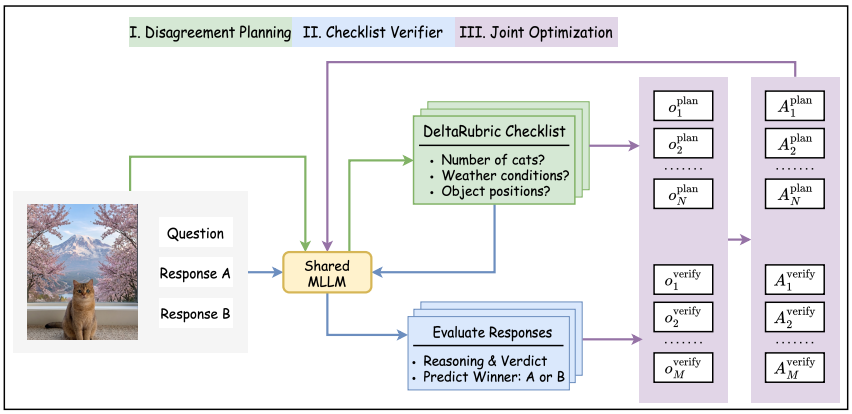


Figure 1: **Overview of DeltaRubric.** Given an input tuple (I, q, y_A, y_B), a single shared MLLM operates in two sequential roles. As a *Disagreement Planner*, the model analyzes the candidate responses to identify critical factual divergences and generates a neutral, instance-specific verification checklist. Example generated checklists are shown in Figure 3 and Appendix A.16. Conditioned on this self-generated checklist, the same model then acts as a *Checklist Verifier*, executing each item against the image I and question q to produce a grounded final judgment. We formulate DeltaRubric as a multi-role RL problem where planning and verification are optimized jointly.

claims and visual evidence. Existing multimodal reward models largely adopt a single-step paradigm, directly mapping inputs to a holistic preference or rationale. However, this single-step evaluation can suffer from lazy judging, a phenomenon where models bypass the demanding task of fine-grained decisions (Zheng et al., 2023). Instead, they exploit textual priors or length biases, failing to rigorously verify the response against the image context (Singhal et al., 2023; Huang et al., 2024). Furthermore, such evaluation often fails to capture the multifaceted nature of response quality, especially in non-verifiable domains (Xu et al., 2026). We argue that this formulation is limited: multimodal evaluation should not be treated as a passive scoring task, but rather an active reasoning process.

Although rubric-based evaluation has proven effective at mitigating these issues in text-only tasks, it remains largely underexplored in the multimodal domain. The primary bottleneck is the complexity of visual reasoning: the critical differences between two multimodal responses often depend on highly specific, instance-level visual details, such as object counts, spatial relationships, or localized hallucinations (Yu et al., 2024). Consequently, multimodal evaluation demands an active mechanism capable of dynamically synthesizing instance-specific rubrics that isolate the critical factual and spatial discrepancies between responses. This limitation leads to a crucial research question: *How can we transform multimodal preference evaluation from a single-step, holistic judgment into a structured, disagreement-driven verification process?*

To answer this, we introduce **DeltaRubric**, a framework that structurally decomposes multimodal evaluation into a sequential, two-step process within a single shared MLLM, as illustrated in Figure 1. Rather than mapping multimodal inputs directly to a verdict, we reformulate evaluation as a plan-and-execute procedure, where the model first induces an explicit verification structure and then executes it for judgment. First, acting as a *Disagreement Planner*, the model analyzes two candidate responses to isolate critical factual divergences, generating a neutral, actionable, and instance-specific verification checklist. Second, transitioning into a *Checklist Verifier*, the model executes each item on the checklist against the visual evidence, aggregating the grounded findings to reach a final judgment.

Training a single model to perform both planning and verification introduces a key challenge: how to jointly optimize planning quality and verification accuracy? To address this, we formulate DeltaRubric as a multi-role reinforcement learning problem, where planning and verification are optimized with distinct yet coordinated objectives. The Planner is rewarded for generating rubric checklists that expose and correct the Verifier's blind spots, while the Verifier is rewarded for accurate and grounded execution. Inspired by recent generative reward modeling paradigms (Xu et al., 2026; Shao et al., 2025), we move beyond static scalar rewards and instead optimize the model's evaluative reasoning process itself. Using group-based RL algorithms such as GRPO (Shao et al., 2024) and DAPO Yu et al. (2025a), we compute task-specific advantages and update both capabilities through a shared policy. This design enables the model to internalize evaluation as a structured, verification-

driven reasoning process, resulting in a robust generative reward model that generalizes across complex multimodal tasks.

We validate our approach by training Qwen3-VL 4B and 8B Instruct (Bai et al., 2025) models and evaluating them on a comprehensive benchmark suite. On VL-RewardBench (Li et al., 2025a), DeltaRubric improves overall accuracy of base models by +22.6 (4B) and +18.8 (8B) points, and consistently outperforms the no-rubric baselines (+4.3 and +8.1, respectively). On Multimodal RewardBench (Yasunaga et al., 2025), it improves the overall accuracy of the 8B base model by +5.5 and surpasses the no-rubric baseline by +4.5. Furthermore, we evaluate on the text-only RewardBench (Lambert et al., 2025), DeltaRubric elevates the 8B base model's overall accuracy by +3.2, indicating that multimodal finetuning with DeltaRubric preserves, and even enhances foundational language capabilities. Beyond benchmark accuracy, DeltaRubric surpasses compute-matched inference scaling (verbose CoT and best-of-N) while using fewer tokens, outperforms DPO and prompt optimization, generalizes to a different architecture InternVL3 (Zhu et al., 2025) and to scalar rewards, and as a frozen reward model improves downstream GRPO policy training. Overall, decomposing evaluation into structured, verifiable steps yields more reliable and generalizable multimodal reward modeling. In summary, our work offers the following key contributions:

- We propose DeltaRubric, a novel approach that reframes multimodal evaluation as an active, two-step visual investigation.
- By decoupling the evaluation process into a Planner and Verifier, optimized jointly via multi-role RL, DeltaRubric encourages the model to isolate factual contradictions and ground its judgments in visual evidence, effectively mitigating lazy judging and improving evaluation reliability.
- DeltaRubric achieves solid empirical gains. On VL-RewardBench, it improves base model overall accuracy by **+22.6** (4B) and **+18.8** (8B) points, largely outperforming no-rubric, compute-matched, preference-learning (DPO), and prompt-optimization baselines while using fewer tokens. Text-only RewardBench evaluations further show that DeltaRubric prevents catastrophic forgetting while enhancing foundational reasoning, and as a frozen reward model it improves downstream GRPO policy training.

2 Related Work

Multimodal Reward Modeling. The alignment of MLLMs heavily relies on extending RLHF to the visual domain, necessitating robust multimodal reward models (Sun et al., 2024; Yu et al., 2024). Early efforts primarily adapted the LLM-as-a-judge paradigm (Zheng et al., 2023; Chen et al., 2024a) to evaluate the interaction between textual claims and visual inputs (Xiong et al., 2025; Yu et al., 2025b). Recently, progress has been made in both optimizing direct scalar reward baselines (Zhang et al., 2026) and developing generative multimodal reward models that incorporate CoT reasoning to improve reliability (Wang et al., 2025; Zhang et al., 2025a; Wang et al., 2026). Regardless of the specific architecture, most methods train a monolithic model to process visual inputs and output either a direct preference score or a holistic rationale (Zhang et al., 2025b).

Despite these advancements, monolithic multimodal evaluators exhibit biases similar to the lazy judging phenomenon observed in text-based LLMs (Zheng et al., 2023). Because fine-grained visual grounding is inherently challenging, models often bypass rigorous image verification and instead exploit language priors, formatting, or length biases (Singhal et al., 2023; Huang et al., 2024). While recent work has attempted to reinforce visual reasoning via agentic tool use (Ding et al., 2025), current methods still lack an intrinsic mechanism that enforces explicit visual investigation. Our framework, DeltaRubric, bridges this gap by shifting evaluation from a passive scoring task to an active, two-step process. By structurally decoupling the isolation of contested textual claims (the Planner) from the grounded verification against visual evidence (the Verifier), DeltaRubric neutralizes textual bias and enforces structured, instance-level visual verification.

Rubrics as Rewards. To address the opacity and unreliability of direct preference scoring in open-ended, non-verifiable tasks, the text-only domain has increasingly adopted rubric-based and checklist-driven evaluation frameworks (Viswanathan et al., 2025; Gunjal et al., 2025). By decomposing complex judgments into explicitly defined criteria, these methods reduce cognitive load and improve reward model alignment (Huang et al., 2025; Liu et al., 2025c). Recent approaches have further scaled these concepts, utilizing alternating reinforcement learning and self-evolving rubrics to reinforce CoT reasoning, handle deep research, and guide non-verifiable post-training (Sheng et al., 2026; Shao et al., 2025; Xu et al., 2026).

However, the application of rubric-based rewards to the multimodal domain remains largely underexplored. Unlike text evaluation, visual evaluation requires verifying highly specific, instance-level physical realities, such as localized hallucinations, object counts, and spatial relationships (Yu et al., 2024). While recent work has begun exploring rubric-based generative rewards for multimodal reasoning (Jia et al., 2025), existing pipelines rely on disjointed architectures prone to cascading errors. Our approach addresses this by dynamically synthesizing disagreement-focused rubrics directly from candidate conflicts. Furthermore, unlike previous approaches that rely on separate models for rubric generation and preference evaluation (Xu et al., 2026), DeltaRubric jointly optimizes both capabilities via multi-role reinforcement learning. The decoupled advantage estimation ensures the model learns to actively hunt for critical visual discrepancies without cross-task variance corrupting learning.

3 APPROACH

We present **DeltaRubric**, a framework for multimodal reward modeling that decomposes evaluation into a self-guided, two-step process within a single shared MLLM. Instead of directly predicting a scalar reward or binary verdict from a visual prompt, DeltaRubric first generates a disagreement-focused verification checklist (the *Planner*) and then executes this checklist against the image to derive the final judgment (the *Verifier*). Both roles are jointly optimized through multi-role RL.

3.1 PROBLEM FORMULATION

Let each training sample be denoted as $x = (I, q, y_{\mathrm{A}}, y_{\mathrm{B}})$, where I is the image, q is the question, and $y_{\mathrm{A}}, y_{\mathrm{B}}$ are two candidate responses, with $z^* \in \{\mathrm{A}, \mathrm{B}\}$ representing the preferred response. The objective is to predict the superior response $z \in \{\mathrm{A}, \mathrm{B}\}$. Standard RLHF approaches directly model $\pi_\theta(z \mid x)$ or generate reasoning before prediction $\pi_\theta(r, z \mid x)$. However, such single-step evaluation is prone to lazy judging, where models rely on textual priors or superficial patterns instead of grounded visual verification. To address this, we reformulate multimodal evaluation as a latent plan-generation and execution problem mediated by an intermediate, self-generated verification checklist c.

3.2 PLANNER-VERIFIER ARCHITECTURE

To enforce fine-grained and grounded evaluations, we introduce a shared policy model π_θ that acts consecutively as a *Planner* and a *Verifier*.

Disagreement Planner. Given an input tuple $x = (I, q, y_A, y_B)$, the Planner generates a checklist $c \sim \pi_\theta(\cdot \mid x)$. The checklist consists of a short sequence of verifiable constraints (e.g., concrete visual attributes, object counts, spatial relations, or hallucinated claims) identifying exactly where the two candidate responses fundamentally disagree. We prompt the model to output a strictly neutral, evidence-seeking checklist without expressing a preference for either candidate. A deterministic post-generation filter further enforces this impartiality by rejecting checklists that name a candidate or assert a verdict, in which case the item is judged without a checklist (Appendix A.13). Generated checklist examples and the prompt template are provided in Appendix A.16 and A.17.

Checklist Verifier. The Verifier takes the original input along with the generated checklist to produce the final evaluation. It generates a step-by-step reasoning trajectory r followed by the final verdict z: $(r, z) \sim \pi_\theta(\cdot \mid x, c)$. The Verifier explicitly evaluates each item on the checklist against the image I before aggregating the evidence to decide the winner. The Verifier is instructed to treat the checklist as a shortlist of checks to execute, ignoring any checks that are hallucinated or contradicted by the image. The prompt template for verifier evaluation can be found in Appendix A.17.

3.3 JOINT OPTIMIZATION VIA MULTI-ROLE RL

We jointly optimize the Planner and Verifier capabilities of the shared model π_θ through a multi-role reinforcement learning objective. During each training iteration, π_θ performs both tasks sequentially, generating distinct sets of rollouts for planning and verification. Crucially, the advantages for the Planner and the Verifier are computed independently within their respective task groups. This decoupled advantage estimation allows the isolated signals to be aggregated into a single, unified joint loss function for the final policy update, effectively preventing cross-task variance.

Planner Learning. For a given input x, we sample N candidate checklists $\mathcal{C} = \{c_1, \ldots, c_N\} \sim \pi_\theta(\cdot \mid x)$ from the current policy. To efficiently score each checklist c_i, we query the Verifier using a lightweight cheap probe prompt to obtain a fast verdict without extended reasoning: $z_i \sim \pi_\theta(z \mid x, c_i)$.

The cheap probe prompt template is provided in Appendix A.17, and we verify in Appendix A.12 that the probe accuracy closely tracks that of the full Verifier. Concurrently, we obtain a baseline verdict without providing any rubric checklist: $z_0 \sim \pi_\theta(z \mid x)$. The planner reward is defined by its relative ability to improve over the baseline accuracy:

$$R_{plan}(c_i) = \mathbb{I}(z_i = z^*) - \mathbb{I}(z_0 = z^*), \tag{1}$$

where $\mathbb{I}(\cdot)$ is the indicator function. Therefore, a checklist receives $+1$ reward if it flips an incorrect no-rubric baseline verdict to correct, -1 if it misleads the verifier into an error, and 0 otherwise. Notably, a checklist that merely restates the model's own unguided verdict earns zero reward; positive reward requires evidence that overturns the shared model's prior, so the Planner is never rewarded for justifying a preferred answer, guarding against confirmation bias despite the shared parameters. We then calculate the Planner advantage $A^{(i)}_{plan}$ by normalizing the rewards within the group of N candidate checklists: $A^{(i)}_{plan} = \frac{R_{plan}(c_i) - \mu_{\mathcal{C}}}{\sigma_{\mathcal{C}}}$, where $\mu_{\mathcal{C}}$ and $\sigma_{\mathcal{C}}$ are the mean and standard deviation of $\{R_{plan}(c_1), \ldots, R_{plan}(c_N)\}$.

Verifier Learning. After scoring the Planner candidates, we run a greedy forward pass through the Planner: $c^* = \arg\max \pi_\theta(\cdot \mid x)$ and pass this greedy checklist to the Verifier. We then sample M full reasoning trajectories and verdicts $\mathcal{R} = \{(r_1, z_1), \ldots, (r_M, z_M)\} \sim \pi_\theta(\cdot \mid x, c^\star)$ from the Verifier conditioned on c^*. The Verifier is rewarded based on final accuracy and a conditional guidance bonus:

$$R_{verify}(r_j, z_j) = \mathbb{I}(z_j = z^*) + \lambda \max\left(0, \mathbb{I}(z_j = z^*) - \mathbb{I}(z_0 = z^*)\right). \tag{2}$$

Here, the final accuracy term rewards correct verdicts, while the bonus term specifically rewards cases where checklist-guided verification strictly improves upon the no-guidance baseline, enforced by a $\max(0, \cdot)$ threshold. Similarly, the Verifier advantage $A^{(j)}_{verify}$ is normalized strictly within the M verifier trajectories generated for that prompt: $A^{(j)}_{verify} = \frac{R_{verify}(r_j, z_j) - \mu_{\mathcal{R}}}{\sigma_{\mathcal{R}}}$.

Joint Multi-Role Loss. The final policy update combines the separate experiences into a single optimization step. Let $\mathcal{L}(\theta \mid \text{data}, A)$ denote the standard RL clipped surrogate objective (e.g., GRPO). The shared model θ is updated by minimizing the joint loss:

$$\mathcal{L}_{total}(\theta) = \mathbb{E}_{x \sim \mathcal{D}} \left[\frac{1}{N} \sum_{i=1}^{N} \mathcal{L}\left(\theta \mid c_i, A^{(i)}_{plan}\right) + \frac{1}{M} \sum_{j=1}^{M} \mathcal{L}\left(\theta \mid r_j, z_j, A^{(j)}_{verify}\right) \right]. \tag{3}$$

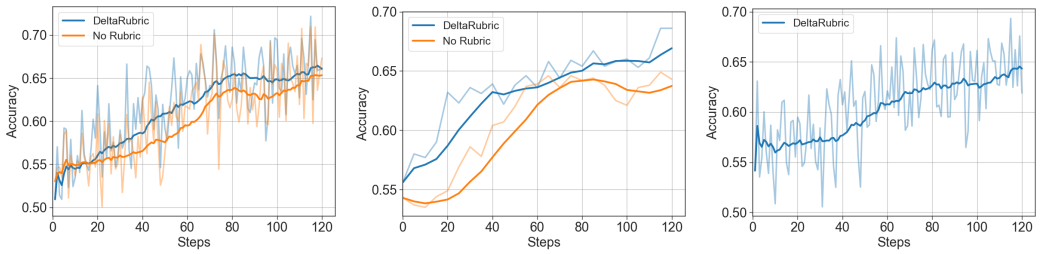
By computing advantages separately for each task group, we ensure that the Planner gradients are strictly driven by checklist quality, and Verifier gradients are strictly driven by execution quality, preventing cross-task variance from corrupting the RL signals. Sharing a single policy further lets the Planner co-adapt to the exact Verifier that executes its checklists, which is what Eq. 1 rewards, and requires serving only one model at inference.

4 Experiments

4.1 Experimental Setup

Implementation Details. We conduct direct RL fine-tuning on the Qwen3-VL-4B and 8B Instruct models (Bai et al., 2025), utilizing GRPO as the underlying RL algorithm. During training, we sample N $= 5$ candidate checklists per prompt for the Planner, and M $= 5$ reasoning trajectories per prompt for the Verifier; see Appendix A.4 for sensitivity. For the verifier reward defined in Eq. 2, we set the guidance bonus coefficient to $\lambda = 0.4$; see Appendix A.3 for sensitivity. Our implementation is built on the EasyR1 framework (Zheng et al., 2025b). More details are provided in Appendix A.1.

Dataset and Benchmarks. To construct the training dataset, we randomly sample 30K instances from the RLAIF-V dataset (Yu et al., 2025b). Each instance consists of an image-query pair, two candidate responses, and a preference label. We strictly decontaminate this data to ensure zero overlap with our evaluation sets. We validate our approach on rigorous benchmarks: VL-RewardBench (Li et al., 2025a), an out-of-domain set designed to probe robustness to common failure modes such as visual hallucinations and spatial reasoning errors; and Multimodal RewardBench (Yasunaga et al., 2025), which evaluates general vision-language preference alignment.



(a) Verifier training. (b) Verifier validation. (c) Planner probe accuracy.

Figure 2: **Planner and Verifier training dynamics. (a)** and **(b)** compare the Verifier's training and validation accuracy (evaluated every 5 steps) against a no-rubric baseline. While both improve, DeltaRubric achieves a higher mean accuracy. **(c)** tracks the Planner probe accuracy, the fraction of sampled checklists that guide a lightweight verdict probe to the correct winner; its steady rise confirms the Planner learns increasingly decision-useful checklists.

Baselines. For our controlled baseline comparisons, we evaluate: (1) a zero-shot base model, where off-the-shelf models are prompted to act as a judge without any RL fine-tuning; (2) a no-rubric setting, where RL-finetuned models generate a CoT rationale followed by a verdict, representing a standard reward modeling paradigm; and (3) DPO (Rafailov et al., 2023), a Bradley–Terry scalar reward model (Bradley and Terry, 1952), GEPA prompt optimization (Agrawal et al., 2026), and compute-matched inference scaling (Section 4.4). In addition, for broader context, we include evaluated results from several external models, including SliME (Zhang et al., 2024), VITA-1.5 (Fu et al., 2025), Molmo-7B (Deitke et al., 2025), InternVL2/3-8B (Chen et al., 2024c; Zhu et al., 2025), Llama-3.2 (Grattafiori et al., 2024), MM-RLHF-Reward-7B (Zhang et al., 2025b), LLaVA-Critic-8B (Xiong et al., 2025), and NVLM-D-72B (Dai et al., 2024).

4.2 Main Results

We illustrate the training dynamics of the Planner and Verifier in Figure 2. While both DeltaRubric and the no-rubric baseline improve over time, DeltaRubric achieves higher Verifier accuracy in both training (Figure 2a) and validation (measured every five steps, Figure 2b). Figure 2c plots the Planner probe accuracy, measured as the fraction of sampled checklists that successfully guide a lightweight verdict probe to the correct ground-truth winner, which serves as a proxy for checklist quality. Its steady increase, aligned with the Verifier gains, indicates that the generated checklists become progressively more decision-useful over training, highlighting the effectiveness of DeltaRubric.

We then present the evaluation results on VL-RewardBench in Table 1. Following the evaluation protocol of (Li et al., 2025a), we compute accuracy via greedy decoding. We report subcategory accuracy (the proportion of correct predictions within each subset), overall accuracy (performance across the entire dataset), and the macro-average (the mean of all subcategory accuracies). As shown, while applying standard preference optimization without rubrics improves upon the base capabilities of both the Qwen3-VL 4B and 8B Instruct models, DeltaRubric drives larger gains. Specifically, our framework outperforms the no-rubric baseline in overall accuracy by 4.3 (4B) and 8.1 (8B) points.

Consequently, our approach achieves the best performance across all evaluation aspects of the benchmark. By explicitly forcing the model to generate a targeted disagreement checklist prior to evaluation, the Planner isolates the exact attributes that distinguish the two candidate responses, while the Verifier grounds the final verdict in visual evidence, mitigating lazy judging.

We then evaluate on the Multimodal RewardBench. We report the overall and subcategory accuracy in Table 2, following the evaluation protocol of (Yasunaga et al., 2025). The results across both Qwen3-VL-4B and 8B models demonstrate a clear benefit to our proposed approach. For the 8B model, standard preference optimization without rubrics achieves a baseline accuracy of 68.7. Our approach improves the overall accuracy to 73.2, validating that instance-specific and disagreement-focused rubrics help capture the complex visual nuances of multimodal prompts.

Analyzing the subcategories reveals exactly where this active visual investigation excels. Our approach yields large gains in heavily visually-dependent tasks such as VQA and Safety. For the 8B model, VQA accuracy improves from a base of 71.4 to 84.9, while Safety accuracy reaches

Table 1: **Evaluation on the VL-RewardBench.** DeltaRubric improves the overall accuracy of the Qwen3-VL-4B and 8B Instruct base models by **+22.6** and **+18.8** points, and consistently outperforms the standard no-rubric baseline (**+4.3** for 4B and **+8.1** for 8B) and other baseline methods.

Models	General	Hallucination	Reasoning	Overall	Macro Avg
Open-Source Models					
VITA-1.5-7B	18.6	8.9	22.1	16.5	16.5
SliME-7B	7.2	27.1	18.6	19.0	17.6
Molmo-7B	31.1	31.8	56.2	37.5	39.7
MM-RLHF-Reward-7B	45.0	50.5	57.6	50.2	51.0
InternVL2-8B	35.6	41.1	59.0	44.5	45.2
LLaVA-Critic-8B	54.6	38.3	59.1	41.2	50.7
Llama-3.2-11B	33.3	38.4	56.6	42.9	42.8
NVLM-D-72B	38.9	31.6	62.0	40.1	44.1
Llama-3.2-90B	42.6	57.3	61.7	56.2	53.9
DeltaRubric					
Qwen3-VL-4B Instruct	46.4	65.0	36.0	54.9	49.1
+ No rubric	51.9	87.9	50.8	73.2	63.5
+ DeltaRubric	55.3	87.8	65.9	77.5	69.7
Qwen3-VL-8B Instruct	47.0	72.4	43.2	61.3	54.2
+ No rubric	55.8	86.1	48.3	72.0	63.4
+ **DeltaRubric**	**59.7**	**88.3**	**72.6**	**80.1**	**73.5**

Table 2: **Evaluation on the Multimodal RewardBench.** DeltaRubric improves the overall accuracy of the Qwen3-VL-8B Instruct base model by **+5.5** points and outperforms the standard no-rubric baseline by **+4.5** points, demonstrating the effectiveness of our approach.

Model	Overall	General		Knowledge	Reasoning		Safety	VQA
		Correctness	Preference		Math	Coding		
Open-Source Models								
VITA-1.5-7B	53.6	55.6	54.3	52.5	51.9	52.8	58.1	50.0
Molmo-7B	52.9	56.8	59.4	54.6	50.7	53.4	34.8	60.3
MM-RLHF-Reward-7B	67.1	61.7	67.5	54.3	58.4	57.9	92.9	76.8
SliME-8B	42.0	42.3	52.2	47.5	43.5	35.3	19.1	53.8
InternVL3-8B	63.6	59.6	61.6	60.5	65.1	56.6	59.3	82.3
Llama-3.2-11B	51.2	57.8	65.8	55.5	50.6	51.7	20.9	55.8
Llama-3.2-90B	61.2	60.0	68.4	61.2	56.3	53.1	52.0	77.1
DeltaRubric								
Qwen3-VL-4B Instruct	65.3	66.1	56.3	52.7	46.7	54.4	80.4	70.8
+ No rubric	66.4	74.5	60.1	59.4	54.3	55.0	87.6	71.3
+ DeltaRubric	69.1	73.7	65.4	60.0	58.0	52.0	91.2	80.8
Qwen3-VL-8B Instruct	67.7	68.9	61.5	56.2	64.6	49.6	82.6	71.4
+ No rubric	68.7	75.0	62.7	56.7	64.0	51.5	91.5	77.0
+ **DeltaRubric**	**73.2**	**76.9**	**65.9**	**69.5**	**68.7**	**52.6**	**93.3**	**84.9**

93.3. This confirms that synthesizing targeted rubrics effectively neutralizes the lazy judging and textual priors that bottleneck traditional monolithic evaluation methods. Qualitative examples in Appendix A.16 (e.g., Figure 3) show that while the no-rubric baseline misses hallucinated details, DeltaRubric generates a targeted disagreement checklist that enforces visual verification and catches the hallucination.

4.3 Ablation Studies

To rigorously validate our design and training configurations, we conduct a comprehensive series of ablation studies. Specifically, we analyze the impact of Planner optimization and evaluate different Planner reward formulations. We then compare our dynamic approach against a static rubric baseline, and verify the preservation of unimodal capabilities on text-only benchmarks.

Ablation on Planner Optimization. We investigate the necessity of active rubric optimization by testing a configuration where the Planner is not fine-tuned via RL. In this setting, the Planner operates entirely zero-shot, relying strictly on greedy decoding to generate a static checklist, while the Verifier is trained independently to execute it. As shown in Table 3, although this frozen configuration provides baseline structural guidance and improves overall accuracy (78.8), removing the Planner's learning signal bottlenecks the framework's capability. Specifically, the untrained Planner struggles to dynamically adapt to complex reasoning, yielding only 66.3 in the Reasoning subcategory. By contrast, our fully trained framework achieves a **+6.3** point improvement in Reasoning (72.6) compared to the frozen Planner. This demonstrates that multi-task RL is essential for teaching the Planner to synthesize targeted, instance-specific rubrics. We further replace our Planner with GPT-5.5 (OpenAI, 2026), which generates checklists offline with the same prompt and filter for our trained Verifier to execute: DeltaRubric performs comparably to this variant, showing that our 8B Planner can approach the rubric-generation capability of a substantially stronger external model (Appendix A.11).

Table 3: **Ablation on Planner optimization and reward.** VL-RewardBench, 8B. Training the Planner via RL drives a **+6.3** Reasoning gain over a frozen Planner, and our relative reward (Eq. 1) yields **+2.5** Overall and **+3.5** Reasoning gains over an unbaselined absolute reward (Abs.).

Models	Gen.	Hallu.	Reas.	All	Macro
Base 8B	47.0	72.4	43.2	61.3	54.2
Ours (Frozen)	58.6	**89.0**	66.3	78.8	71.3
Ours (Abs.)	58.6	85.9	69.1	77.6	71.2
Ours (Trained)	**59.7**	88.3	**72.6**	**80.1**	**73.5**

Ablation on Planner Reward Formulation. To validate our choice of the relative improvement reward for the Planner (Eq. 1), we evaluate an alternative formulation where the Planner is rewarded strictly based on the absolute accuracy of the final verdict ($R_{plan}(c_i) = \mathbb{I}(z_i = z^*)$). As shown in Table 3, this formulation degrades performance across all metrics compared to our relative reward. Most notably, the Overall accuracy drops from 80.1 to 77.6 and the Reasoning score reduces from 72.6 to 69.1. This degradation shows the flaw of the unbaselined approach: it fails to account for instances where the model can easily predict the ground truth without any structural guidance. Consequently, the Planner receives positive reinforcement even for generating irrelevant checklists on trivial tasks, leading to noisy gradient updates. This confirms that the Planner should be explicitly incentivized to correct the Verifier's blind spots, rather than merely sharing credit for easy successes.

Comparison with a Static Rubric Baseline. To isolate the value of generating instance-specific checklists, we evaluate a static-rubric baseline. In this setting, the RL-finetuned Verifier is conditioned on a generic, dataset-level evaluation prompt (e.g., "check for hallucinations, logical consistency, and correct object identification") rather than a dynamically generated checklist. Please see the static-rubric prompt template in Appendix A.17. As shown in Table 4, providing a generic structural prior does offer an improvement over the no-rubric baseline (raising 8B Overall accuracy from 72.0 to 75.3). However, a static prompt fundamentally lacks the capacity to adapt to diverse, instance-specific visual contradictions. Consequently, our dynamic framework (DeltaRubric) largely outperforms the static baseline across all metrics. Most notably, dynamically generating the rubric yields a **+13.0** point gain in the Reasoning subcategory for the 8B model. This confirms that actively synthesizing a targeted checklist to guide evaluation is beneficial.

Table 4: **Ablation with a static rubric baseline.** Performance on the VL-RewardBench. While a static rubric provides a useful structural prior, dynamically generating instance-specific checklists drives a large **+13.0** point gain in the Reasoning subcategory for the 8B model.

Models	Gen.	Hallu.	Reas.	All	Macro
Base 4B	46.4	65.0	36.0	54.9	49.1
No rubric	51.9	87.9	50.8	73.2	63.5
Static rubric	47.5	85.7	62.8	74.3	65.3
Ours	**55.3**	**87.8**	**65.9**	**77.5**	**69.7**
Base 8B	47.0	72.4	43.2	61.3	54.2
No rubric	55.8	86.1	48.3	72.0	63.4
Static rubric	53.6	87.2	59.6	75.3	66.8
Ours	**59.7**	**88.3**	**72.6**	**80.1**	**73.5**

Evaluation on Text-Only Benchmark. To evaluate whether multimodal fine-tuning degrades pre-existing language capabilities, we benchmark our models on the text-only RewardBench (Lambert et al., 2025) (Table 11, Appendix A.6). The base Qwen3-VL-8B Instruct model achieves a strong 81.4 overall accuracy despite its relatively low multimodal performance, suggesting a clear modality gap. Crucially, multimodal fine-tuning with DeltaRubric not only avoids catastrophic forgetting, but further improves overall accuracy to 84.6, with the largest gain in the Reasoning subset (79.0

to 85.7). At the same time, the substantially larger gains on multimodal benchmarks indicate that DeltaRubric specifically improves grounded visual verification rather than merely general reasoning.

4.4 Comparison with Alternative Baselines and Downstream Alignment

Compute-Matched Baselines. We examine whether the gains of DeltaRubric stem from more inference compute by scaling the RL-trained no-rubric 8B judge with verbose CoT, which instructs the judge to reason in detail and examine each factual claim against the image, and with best-of-3 majority voting (Wang et al., 2023b). As shown in Table 5, both spend 2.8–3.3× the generated tokens yet gain at most 3.8 points, whereas DeltaRubric uses only 1.10× the tokens, as its concise checklists (97 tokens on average) shorten the verification, and outperforms both by at least 4.3 points (Appendix A.10). The gains thus stem from the learned planning and verification rather than a larger token budget.

Table 5: **Comparison with alternative baselines.** VL-RewardBench Overall/Macro (Qwen3-VL-8B); GEPA uses frozen base weights, parentheses give tokens relative to the no-rubric judge. DeltaRubric surpasses all these baselines at comparable token cost, so its gains stem from learned planning and verification, not prompt engineering or extra compute (Appendix A.8–A.10).

Method	All	Macro
Base 8B	61.3	54.2
GEPA, no-rubric prompt	65.0	59.0
GEPA, Planner + Verifier	70.3	64.7
DPO (implicit reward)	68.2	62.8
Bradley–Terry scalar RM	68.6	61.3
No rubric (1.00×)	72.0	63.4
+ Verbose CoT (2.81×)	74.3	66.6
+ Best-of-3 vote (3.28×)	75.8	68.0
DeltaRubric (1.10×)	**80.1**	**73.5**

Alternative Paradigms. With frozen weights, prompt optimization via GEPA (Agrawal et al., 2026), using GPT-5.5 as the reflector, raises the base 8B model from 61.3 to 65.1 Overall for the no-rubric prompt and to 70.3 for jointly optimized Planner and Verifier prompts, far below DeltaRubric (Appendix A.8). With the same backbone and 30K training pairs, DPO (Rafailov et al., 2023), scored by its implicit reward, and a Bradley–Terry scalar reward model (Bradley and Terry, 1952) reach 68.2 and 68.6, and neither produces a rationale. Conditioning the scalar model on DeltaRubric checklists improves it to 72.2, showing that our formulation also extends to scalar rewards (Appendix A.9).

Downstream Policy Training. To test whether reward model gains translate into better policies, we train Qwen3-VL-8B Instruct policies with GRPO for 120 steps on 17.4K RLAIF-V prompts disjoint from the reward model training data, using either the no-rubric or the DeltaRubric 8B reward model as the frozen reward. As GRPO normalizes advantages within a group, pairwise verdicts are used directly by comparing sampled responses against an anchor response (Appendix A.15). As shown in Table 6, the DeltaRubric-trained policy outperforms the no-rubric one on hallucination (AMBER (Wang et al., 2023a), MMHal (Sun et al., 2024)) and general capability (MMStar (Chen et al., 2024b)) benchmarks, showing that reward model gains transfer to downstream post-training.

Table 6: **Downstream policy training.** Qwen3-VL-8B policies trained with GRPO, differing only in the reward model (RM). Higher is better.

Policy	AMBER	MMHal	MMStar
Base policy	87.1	3.18	59.8
+ No-rubric RM	88.7	3.56	61.2
+ DeltaRubric RM	**90.9**	**4.14**	**63.7**

Robustness and Faithfulness. DeltaRubric generalizes to DAPO (Yu et al., 2025a) (Appendix A.5), is robust to N and M (Appendix A.4), and substantially outperforms the no-rubric baseline when trained on a different architecture, improving InternVL3-2B (Zhu et al., 2025) from 41.7 to 59.5 Overall (Appendix A.7). To test whether the Verifier faithfully executes the checks rather than pattern-matching on their presence, we negate the decisive checklist item with GPT-5.5: Overall accuracy drops from 80.1 to 64.4, below the no-rubric judge (Appendix A.14).

5 Conclusions

In this work, we introduce DeltaRubric, an approach that reframes multimodal evaluation as an active, two-step visual investigation. By decomposing evaluation into a *Disagreement Planner* and a *Checklist Verifier* optimized via multi-role reinforcement learning, the model isolates factual contradictions and grounds its judgments in visual evidence rather than textual priors, mitigating lazy judging. DeltaRubric substantially outperforms no-rubric, static-rubric, compute-matched, and prompt-optimization baselines, its gains transfer to downstream policy training, and text-only RewardBench results show it avoids catastrophic forgetting while improving core reasoning.

REFERENCES

Lakshya A Agrawal, Shangyin Tan, Dilara Soylu, Noah Ziems, Rishi Khare, Krista Opsahl-Ong, Arnav Singhvi, Herumb Shandilya, Michael J Ryan, Meng Jiang, Christopher Potts, Koushik Sen, Alex Dimakis, Ion Stoica, Dan Klein, Matei Zaharia, and Omar Khattab. GEPA: Reflective prompt evolution can outperform reinforcement learning. In *The Fourteenth International Conference on Learning Representations*, 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*, 2025.

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

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

Dongping Chen, Ruoxi Chen, Shilin Zhang, Yaochen Wang, Yinuo Liu, Huichi Zhou, Qihui Zhang, Yao Wan, Pan Zhou, and Lichao Sun. Mllm-as-a-judge: Assessing multimodal llm-as-a-judge with vision-language benchmark. In *Forty-first International Conference on Machine Learning*, 2024a.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, and Feng Zhao. Are we on the right way for evaluating large vision-language models? In *Advances in Neural Information Processing Systems*, volume 37, 2024b.

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24185–24198, 2024c.

Runpeng Dai, Linfeng Song, Haolin Liu, Zhenwen Liang, Dian Yu, Haitao Mi, Zhaopeng Tu, Rui Liu, Tong Zheng, Hongtu Zhu, et al. Cde: Curiosity-driven exploration for efficient reinforcement learning in large language models. *arXiv preprint arXiv:2509.09675*, 2025.

Wenliang Dai, Nayeon Lee, Boxin Wang, Zhuolin Yang, Zihan Liu, Jon Barker, Tuomas Rintamaki, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. Nvlm: Open frontier-class multimodal llms. *arXiv preprint arXiv:2409.11402*, 2024.

DeepSeek-AI. Deepseek-v4: Towards highly efficient million-token context intelligence, 2026.

Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Mohammadreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, et al. Molmo and pixmo: Open weights and open data for state-of-the-art vision-language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 91–104, 2025.

Shengyuan Ding, Xinyu Fang, Ziyu Liu, Yuhang Zang, Yuhang Cao, Xiangyu Zhao, Haodong Duan, Xiaoyi Dong, Jianze Liang, Bin Wang, et al. Arm-thinker: Reinforcing multimodal generative reward models with agentic tool use and visual reasoning. *arXiv preprint arXiv:2512.05111*, 2025.

Chaoyou Fu, Haojia Lin, Xiong Wang, Yi-Fan Zhang, Yunhang Shen, Xiaoyu Liu, Haoyu Cao, Zuwei Long, Heting Gao, Ke Li, et al. Vita-1.5: Towards gpt-4o level real-time vision and speech interaction. *arXiv preprint arXiv:2501.01957*, 2025.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Anisha Gunjal, Anthony Wang, Elaine Lau, Vaskar Nath, Yunzhong He, Bing Liu, and Sean Hendryx. Rubrics as rewards: Reinforcement learning beyond verifiable domains. *arXiv preprint arXiv:2507.17746*, 2025.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Qidong Huang, Xiaoyi Dong, Pan Zhang, Bin Wang, Conghui He, Jiaqi Wang, Dahua Lin, Weiming Zhang, and Nenghai Yu. Opera: Alleviating hallucination in multi-modal large language models via over-trust penalty and retrospection-allocation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13418–13427, 2024.

Zenan Huang, Yihong Zhuang, Guoshan Lu, Zeyu Qin, Haokai Xu, Tianyu Zhao, Ru Peng, Jiaqi Hu, Zhanming Shen, Xiaomeng Hu, et al. Reinforcement learning with rubric anchors. *arXiv preprint arXiv:2508.12790*, 2025.

Mengzhao Jia, Zhihan Zhang, Ignacio Cases, Zheyuan Liu, Meng Jiang, and Peng Qi. Autorubric-r1v: Rubric-based generative rewards for faithful multimodal reasoning. *arXiv preprint arXiv:2510.14738*, 2025.

Seungone Kim, Jamin Shin, Yejin Cho, Joel Jang, Shayne Longpre, Hwaran Lee, Sangdoo Yun, Seongjin Shin, Sungdong Kim, James Thorne, et al. Prometheus: Inducing fine-grained evaluation capability in language models. In *The Twelfth International Conference on Learning Representations*, 2023.

Nathan Lambert. Reinforcement learning from human feedback. *arXiv preprint arXiv:2504.12501*, 2025.

Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V Miranda, Alisa Liu, Nouha Dziri, Shane Lyu, et al. Tulu 3: Pushing frontiers in open language model post-training. *arXiv preprint arXiv:2411.15124*, 2024.

Nathan Lambert, Valentina Pyatkin, Jacob Morrison, Lester James Validad Miranda, Bill Yuchen Lin, Khyathi Chandu, Nouha Dziri, Sachin Kumar, Tom Zick, Yejin Choi, et al. Rewardbench: Evaluating reward models for language modeling. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 1755–1797, 2025.

Lei Li, Yuancheng Wei, Zhihui Xie, Xuqing Yang, Yifan Song, Peiyi Wang, Chenxin An, Tianyu Liu, Sujian Li, Bill Yuchen Lin, et al. Vl-rewardbench: a challenging benchmark for vision-language generative reward models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 24657–24668, 2025a.

Zongxia Li, Wenhao Yu, Chengsong Huang, Rui Liu, Zhenwen Liang, Fuxiao Liu, Jingxi Che, Dian Yu, Jordan Boyd-Graber, Haitao Mi, et al. Self-rewarding vision-language model via reasoning decomposition. *arXiv preprint arXiv:2508.19652*, 2025b.

Rui Liu, Dian Yu, Lei Ke, Haolin Liu, Yujun Zhou, Zhenwen Liang, Haitao Mi, Pratap Tokekar, and Dong Yu. Stable and efficient single-rollout rl for multimodal reasoning. *arXiv preprint arXiv:2512.18215*, 2025a.

Rui Liu, Dian Yu, Tong Zheng, Runpeng Dai, Zongxia Li, Wenhao Yu, Zhenwen Liang, Linfeng Song, Haitao Mi, Pratap Tokekar, et al. Vogue: Guiding exploration with visual uncertainty improves multimodal reasoning. *arXiv preprint arXiv:2510.01444*, 2025b.

Tianci Liu, Ran Xu, Tony Yu, Ilgee Hong, Carl Yang, Tuo Zhao, and Haoyu Wang. Openrubrics: Towards scalable synthetic rubric generation for reward modeling and llm alignment. *arXiv preprint arXiv:2510.07743*, 2025c.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL `https://openreview.net/forum?id=Bkg6RiCqY7`.

OpenAI. GPT-5.5 system card. `https://openai.com/index/gpt-5-5-system-card/`, 2026.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

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.

William Saunders, Catherine Yeh, Jeff Wu, Steven Bills, Long Ouyang, Jonathan Ward, and Jan Leike. Self-critiquing models for assisting human evaluators. *arXiv preprint arXiv:2206.05802*, 2022.

Rulin Shao, Akari Asai, Shannon Zejiang Shen, Hamish Ivison, Varsha Kishore, Jingming Zhuo, Xinran Zhao, Molly Park, Samuel G Finlayson, David Sontag, et al. Dr tulu: Reinforcement learning with evolving rubrics for deep research. *arXiv preprint arXiv:2511.19399*, 2025.

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.

Leheng Sheng, Wenchang Ma, Ruixin Hong, Xiang Wang, An Zhang, and Tat-Seng Chua. Reinforcing chain-of-thought reasoning with self-evolving rubrics. *arXiv preprint arXiv:2602.10885*, 2026.

Prasann Singhal, Tanya Goyal, Jiacheng Xu, and Greg Durrett. A long way to go: Investigating length correlations in rlhf. *arXiv preprint arXiv:2310.03716*, 2023.

Zhiqing Sun, Sheng Shen, Shengcao Cao, Haotian Liu, Chunyuan Li, Yikang Shen, Chuang Gan, Liangyan Gui, Yu-Xiong Wang, Yiming Yang, et al. Aligning large multimodal models with factually augmented rlhf. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 13088–13110, 2024.

Luong Trung, Xinbo Zhang, Zhanming Jie, Peng Sun, Xiaoran Jin, and Hang Li. ReFT: Reasoning with reinforced fine-tuning. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7601–7614, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.410. URL https://aclanthology.org/2024.acl-long.410/.

Vijay Viswanathan, Yanchao Sun, Shuang Ma, Xiang Kong, Meng Cao, Graham Neubig, and Tongshuang Wu. Checklists are better than reward models for aligning language models. *arXiv preprint arXiv:2507.18624*, 2025.

Chenglong Wang, Yifu Huo, Yang Gan, Qiaozhi He, Qi Meng, Bei Li, Yan Wang, Junfu Liu, Tianhua Zhou, Jingbo Zhu, et al. Msrl: Scaling generative multimodal reward modeling via multi-stage reinforcement learning. *arXiv preprint arXiv:2603.25108*, 2026.

Junyang Wang, Yuhang Wang, Guohai Xu, Jing Zhang, Yukai Gu, Haitao Jia, Jiaqi Wang, Haiyang Xu, Ming Yan, Ji Zhang, and Jitao Sang. AMBER: An LLM-free multi-dimensional benchmark for MLLMs hallucination evaluation. *arXiv preprint arXiv:2311.07397*, 2023a.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*, 2023b.

Yibin Wang, Zhimin Li, Yuhang Zang, Chunyu Wang, Qinglin Lu, Cheng Jin, and Jiaqi Wang. Unified multimodal chain-of-thought reward model through reinforcement fine-tuning. *arXiv preprint arXiv:2505.03318*, 2025.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

Tianyi Xiong, Xiyao Wang, Dong Guo, Qinghao Ye, Haoqi Fan, Quanquan Gu, Heng Huang, and Chunyuan Li. Llava-critic: Learning to evaluate multimodal models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 13618–13628, 2025.

Ran Xu, Tianci Liu, Zihan Dong, Tony Yu, Ilgee Hong, Carl Yang, Linjun Zhang, Tao Zhao, and Haoyu Wang. Alternating reinforcement learning for rubric-based reward modeling in non-verifiable llm post-training. *arXiv preprint arXiv:2602.01511*, 2026.

Michihiro Yasunaga, Luke Zettlemoyer, and Marjan Ghazvininejad. Multimodal rewardbench: Holistic evaluation of reward models for vision language models. *arXiv preprint arXiv:2502.14191*, 2025.

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*, 2025a.

Tianyu Yu, Yuan Yao, Haoye Zhang, Taiwen He, Yifeng Han, Ganqu Cui, Jinyi Hu, Zhiyuan Liu, Hai-Tao Zheng, Maosong Sun, et al. Rlhf-v: Towards trustworthy mllms via behavior alignment from fine-grained correctional human feedback. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13807–13816, 2024.

Tianyu Yu, Haoye Zhang, Qiming Li, Qixin Xu, Yuan Yao, Da Chen, Xiaoman Lu, Ganqu Cui, Yunkai Dang, Taiwen He, et al. Rlaif-v: Open-source ai feedback leads to super gpt-4v trustworthiness. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 19985–19995, 2025b.

Jiawei Zhang, Tianyu Pang, Chao Du, Yi Ren, Bo Li, and Min Lin. Benchmarking large multimodal models against common corruptions. *arXiv preprint arXiv:2401.11943*, 2024.

Yi-Fan Zhang, Xingyu Lu, Xiao Hu, Chaoyou Fu, Bin Wen, Tianke Zhang, Changyi Liu, Kaiyu Jiang, Kaibing Chen, Kaiyu Tang, et al. R1-reward: Training multimodal reward model through stable reinforcement learning. *arXiv preprint arXiv:2505.02835*, 2025a.

Yi-Fan Zhang, Tao Yu, Haochen Tian, Chaoyou Fu, Peiyan Li, Jianshu Zeng, Wulin Xie, Yang Shi, Huanyu Zhang, Junkang Wu, et al. Mm-rlhf: The next step forward in multimodal llm alignment. *arXiv preprint arXiv:2502.10391*, 2025b.

YiFan Zhang, Haihua Yang, Huanyu Zhang, Yang Shi, Zezhou Chen, Haochen Tian, Chaoyou Fu, Kai WU, Bo Cui, Xu Wang, Jianfei Pan, Haotian Wang, Zhang Zhang, and Liang Wang. Basereward: A strong baseline for multimodal reward model. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=EuN5iszF0a.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623, 2023.

Tong Zheng, Hongming Zhang, Wenhao Yu, Xiaoyang Wang, Runpeng Dai, Rui Liu, Huiwen Bao, Chengsong Huang, Heng Huang, and Dong Yu. Parallel-r1: Towards parallel thinking via reinforcement learning. *arXiv preprint arXiv:2509.07980*, 2025a.

Yaowei Zheng, Junting Lu, Shenzhi Wang, Zhangchi Feng, Dongdong Kuang, Yuwen Xiong, and Richong Zhang. Easyr1: An efficient, scalable, multi-modality rl training framework. https://github.com/hiyouga/EasyR1, 2025b.

Jialun Zhong, Wei Shen, Yanzeng Li, Songyang Gao, Hua Lu, Yicheng Chen, Yang Zhang, Wei Zhou, Jinjie Gu, and Lei Zou. A comprehensive survey of reward models: Taxonomy, applications, challenges, and future. *arXiv preprint arXiv:2504.12328*, 2025.

Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, et al. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025.

A Appendix

A.1 Implementation Details

We perform direct RL training on the Qwen3-VL-4B and 8B Instruct (Bai et al., 2025) models. For the Planner, we sample $N = 5$ candidate checklists per prompt. For the Verifier, we sample $M = 5$ reasoning trajectories per prompt. All models are trained for 120 steps using AdamW (Loshchilov and Hutter, 2019) optimizer with a learning rate of 1×10^{-6} and weight decay of 0.01. We adopt a global batch size of 128, a rollout batch size of 256, and generate rollouts with a temperature 1.0. All models are trained on 8 GPUs. The implementation builds on the EasyR1 framework (Zheng et al., 2025b).

A.2 Training Cost Analysis

We measure the training cost of the no-rubric baseline and DeltaRubric under GRPO with the Qwen3-VL-4B Instruct model, using identical data and hyperparameters (Appendix A.1) for 120 steps on the same 8-GPU node. Table 7 reports the time per training step, the time spent on rollout generation and on the actor update within each step, and the total wall-clock time. DeltaRubric incurs additional overhead compared to the no-rubric baseline due to the Planner-Verifier decomposition, primarily from the extra generation cost of sampling and probing the N candidate checklists, while the actor update cost increases only moderately. This overhead is worth the substantial performance improvements over the baseline.

Table 7: **Training cost.** GRPO training of Qwen3-VL-4B Instruct for 120 steps. DeltaRubric incurs some overhead, but this cost is outweighed by its substantial performance gains over the baseline.

Method	Time/step (s)	Generation (s)	Actor update (s)	Wall-clock (h)
No rubric	102.5	28.1	52.7	3.4
DeltaRubric	150.3	69.6	64.2	5.0

A.3 Sensitivity to the guidance bonus coefficient

We conduct a sensitivity analysis on the guidance bonus coefficient λ for the verifier reward defined in Eq. 2. We test values ranging from $\{0.0, 0.2, 0.4, 0.6\}$, where $\lambda = 0.0$ represents a baseline with no guidance bonus. Table 8 presents the performance of the Qwen3-VL-8B Instruct model trained via our approach on VL-RewardBench. The results indicate that applying a guidance bonus outperforms the unguided baseline ($\lambda = 0.0$), with $\lambda = 0.4$ achieving the optimal balance (Overall Accuracy: 80.1, Macro Avg: 73.5). This confirms that an appropriately scaled bonus effectively incentivizes the verifier to adhere to the generated rubric. However, scaling the coefficient further to $\lambda = 0.6$ results in a degradation, particularly in the reasoning sub-category. This drop suggests that an overly aggressive guidance penalty overrides the primary preference signal, constraining the model's capacity for broader contextual reasoning required to render a correct final verdict.

Table 8: **Sensitivity analysis of the guidance bonus coefficient (λ).** Evaluating our DeltaRubric-trained Qwen3-VL-8B Instruct model on VL-RewardBench reveals that the optimal setting ($\lambda = 0.4$) achieves the highest overall accuracy and macro average. Disabling the bonus ($\lambda = 0.0$) reduces visual grounding, while an excessively high coefficient ($\lambda = 0.6$) degrades complex reasoning by causing the model to over-optimize for checklist adherence.

Coefficients	General	Hallucination	Reasoning	Overall	Macro Avg
$\lambda = 0.0$	55.3	87.9	68.5	78.2	70.6
$\lambda = 0.2$	54.1	86.3	**72.9**	78.2	71.1
$\lambda = 0.4$	**59.7**	**88.3**	72.6	**80.1**	**73.5**
$\lambda = 0.6$	58.6	87.3	66.3	77.8	70.7

A.4 Sensitivity to the Number of Checklists and Verifier Trajectories

We examine the sensitivity of DeltaRubric to the number of candidate checklists N sampled per prompt for the Planner and the number of reasoning trajectories M sampled per prompt for the Verifier. We retrain the Qwen3-VL-8B Instruct model with $N \in \{2, 5, 8\}$ at $M = 5$ and with $M \in \{2, 5, 8\}$ at $N = 5$, keeping all other settings unchanged, and evaluate on VL-RewardBench. As shown in Table 9, performance is stable and robust across settings. Reducing either count to 2 lowers Overall accuracy by only 1.4–1.7 points, mostly in the Reasoning subcategory, as smaller groups provide a less informative group-relative signal. Increasing either count to 8 yields at most $+0.3$ Overall at a higher rollout cost. We therefore adopt $N = M = 5$ as a favorable trade-off between accuracy and training cost.

Table 9: **Sensitivity to N and M.** VL-RewardBench accuracy of the DeltaRubric-trained Qwen3-VL-8B Instruct model with different numbers of sampled checklists (N) and Verifier trajectories (M). Our default setting is $N = M = 5$.

Setting	General	Hallucination	Reasoning	Overall	Macro Avg
$N = 2, M = 5$	58.5	87.6	68.0	78.4	71.4
$N = 5, M = 5$ (default)	59.7	88.3	72.6	80.1	73.5
$N = 8, M = 5$	60.0	88.4	73.0	80.3	73.8
$N = 5, M = 2$	58.8	87.8	68.9	78.7	71.8
$N = 5, M = 8$	60.1	88.5	73.2	80.4	73.9

A.5 Generalization to Alternative RL Algorithm

While our experiments utilize GRPO, the DeltaRubric framework is agnostic to the underlying RL algorithm. To demonstrate this, we evaluate our approach using an alternative algorithm: Decoupled Clip and Dynamic Sampling Policy Optimization (DAPO) (Yu et al., 2025a). Because our decoupled advantage estimation strictly isolates the reward signals for the Planner and the Verifier, DAPO can be seamlessly integrated to govern the joint policy updates. As shown in Table 10 of evaluation on VL-RewardBench, optimizing DeltaRubric with DAPO yields solid gains, outperforming the no-rubric and static-rubric baselines in Overall and Macro accuracy and improving the Overall accuracy of the 4B and 8B base models by 21.0 and 16.6 points, respectively. This reaffirms the effectiveness of our approach.

Table 10: **Evaluation with alternative RL algorithm.** Accuracy on the VL-RewardBench when utilizing DAPO (Yu et al., 2025a) as the underlying RL algorithm. DeltaRubric maintains strong performance advantages over the no-rubric and static-rubric baselines.

Models	General	Hallucination	Reasoning	Overall	Macro Avg
Qwen3-VL-4B Instruct (Bai et al., 2025)	46.4	65.0	36.0	54.9	49.1
+ No rubric	54.1	86.7	56.8	74.3	65.9
+ Static rubric	50.8	86.3	64.7	75.6	67.3
+ DeltaRubric	54.7	86.5	63.1	75.9	68.1
Qwen3-VL-8B Instruct (Bai et al., 2025)	47.0	72.4	43.2	61.3	54.2
+ No rubric	51.4	84.4	63.4	74.2	66.4
+ Static rubric	53.6	83.9	65.3	74.7	67.6
+ DeltaRubric	55.8	87.2	68.8	77.9	70.6

A.6 Evaluation on Text-Only Benchmark

Table 11 reports the full results on the text-only RewardBench (Lambert et al., 2025) discussed in Section 4.3. Multimodal fine-tuning with DeltaRubric not only avoids catastrophic forgetting, but further improves text-only reasoning performance, increasing overall accuracy to 84.6. The largest gain appears in the Reasoning subset, improving from 79.0 to 85.7. These results suggest that disagreement-driven checklist generation encourages the model to decompose evaluations into structured, verifiable reasoning steps, benefiting reasoning even without visual input.

Table 11: **Text-only RewardBench accuracy.** DeltaRubric preserves foundational language capabilities while actively improving structural reasoning (**+6.7**) over the base model.

Models	Chat	Chat Hard	Safety	Reasoning	Overall	Macro Avg
Qwen3-VL-8B Instruct	90.8	70.8	86.8	79.0	81.4	81.9
+ No rubric	89.9	68.4	87.8	80.5	81.9	81.7
+ DeltaRubric	**91.3**	**71.9**	**89.1**	**85.7**	**84.6**	**84.5**

A.7 Architecture Generalization

To examine whether DeltaRubric generalizes beyond the Qwen3-VL family, we train InternVL3-2B (Zhu et al., 2025) with both the no-rubric baseline and DeltaRubric , using the same GRPO training protocol and the same 30K RLAIF-V training pairs as for Qwen3-VL, and evaluate the resulting models on VL-RewardBench. As shown in Table 12, DeltaRubric consistently improves over both the InternVL3-2B base model and the no-rubric baseline, raising Overall accuracy from 41.7 (base) and 48.1 (no rubric) to 59.5, and improving every subcategory. This demonstrates that the rubric-guided planning and verification framework generalizes across architectures.

Table 12: **Architecture generalization.** VL-RewardBench accuracy when training InternVL3-2B with the same GRPO protocol and data as for Qwen3-VL. This demonstrates that our approach generalizes across architectures.

Model	Gen.	Hall.	Reas.	Overall	Macro
InternVL3-2B	31.8	41.0	48.9	41.7	40.6
+ No rubric	33.3	49.7	52.8	48.1	45.3
+ DeltaRubric	**44.5**	**64.2**	**57.0**	**59.5**	**55.2**

A.8 Comparison with Automated Prompt Optimization

Since both the Planner and the Verifier are conditioned on natural-language prompts, we examine whether automated prompt optimization on frozen weights can substitute for RL training. We use GEPA (Agrawal et al., 2026), a reflective prompt optimizer in which a reflector LLM examines failed examples and proposes revised instructions while the model weights remain frozen. We run GEPA with GPT-5.5 (OpenAI, 2026) as the reflector; the model being optimized and evaluated is always the untrained Qwen3-VL-8B Instruct, decoded greedily. The optimization objective is pairwise judging accuracy on a development split drawn from held-out rows of the decontaminated RLAIF-V pool, disjoint from both the 30K RL training pairs and the validation split. VL-RewardBench is never exposed to the optimizer; each resulting prompt is evaluated once on the full benchmark afterwards.

We consider two architectures. For the single-stage no-rubric judge, GEPA rewrites the judge prompt. For the two-stage Planner-Verifier pipeline, we optimize (i) only the Planner prompt, (ii) only the Verifier prompt, (iii) both prompts independently, where GEPA is run separately for each role while holding the other role's prompt at its default, and the two resulting prompts are then deployed together, and (iv) both prompts jointly, where GEPA may revise both prompts and scores every candidate by the pairwise accuracy of the full Planner-Verifier pipeline.

As shown in Table 13, prompt optimization improves performance: GEPA raises the Overall accuracy of the single-stage judge from 61.3 to 65.1. In the two-stage pipeline, optimizing the Planner or Verifier prompt alone improves over the base model, and optimizing both prompts further improves performance, with joint optimization yielding the best GEPA configuration. This indicates that coordinating the Planner-Verifier interaction is beneficial, consistent with our shared-policy design. However, all prompt-optimized configurations remain far below DeltaRubric, which uses the unoptimized prompts in Appendix A.17 and updates only the model weights. While prompt optimization requires only inference calls and is thus much cheaper than RL, it cannot substitute for RL training in our setting.

Table 13: **Comparison with automated prompt optimization.** VL-RewardBench accuracy of Qwen3-VL-8B Instruct. GEPA optimizes the prompts on frozen base weights; DeltaRubric keeps the default prompts and updates the weights via RL.

Method	Gen.	Hall.	Reas.	Overall	Macro
No-rubric judge, default prompt	47.0	72.4	43.2	61.3	54.2
No-rubric judge, GEPA prompt	51.9	74.0	51.2	65.0	59.0
Two-stage, GEPA Planner only	48.6	73.6	46.0	62.9	56.1
Two-stage, GEPA Verifier only	50.8	74.5	50.5	64.9	58.6
Two-stage, GEPA both, independent	51.9	75.8	56.3	67.3	61.3
Two-stage, GEPA both, joint	54.1	77.5	62.5	70.3	64.7
DeltaRubric (RL, default prompts)	**59.7**	**88.3**	**72.6**	**80.1**	**73.5**

A.9 Comparison with Pairwise Preference Learning and Scalar Rewards

Since DeltaRubric is trained on preference pairs, we compare it with standard pairwise preference learning under matched settings. All methods use the Qwen3-VL-8B Instruct backbone and the same 30K decontaminated RLAIF-V training pairs, with the vision tower frozen, and are trained for one epoch using AdamW with a learning rate of 1×10^{-5} and an effective batch size of 128.

- **DPO.** We train with the standard DPO objective (Rafailov et al., 2023) ($\beta = 0.1$), using the frozen base model as the reference policy. At evaluation, each response y receives the implicit reward $r(x, y) = \beta \left[\log \pi_\theta(y \mid x) - \log \pi_{\text{ref}}(y \mid x)\right]$, summed over response tokens, and the higher-scoring response is selected.
- **Bradley–Terry scalar reward model.** To test whether our formulation extends to scalar rewards, we attach a randomly initialized scalar head (learning rate 1×10^{-4}) to the final hidden state of the last token and train a per-response score $s(x, y)$ by maximizing $\log \sigma\left(s(x, y^+) - s(x, y^-)\right)$ (Bradley and Terry, 1952). At evaluation, each response is scored independently and the higher score wins. We further train a checklist-conditioned variant whose input additionally includes the DeltaRubric checklist for the pair, during both training and evaluation.

As shown in Table 14, DPO and the scalar reward model improve over the base model but remain 11.9 and 11.5 points below DeltaRubric in Overall accuracy, and neither produces a rationale for its decision. We note that the two paradigms occupy different roles: DPO consumes preference pairs to produce a policy, whereas DeltaRubric uses them to train a judge whose verdicts can in turn supply the reward for policy training, as shown in Appendix A.15. Moreover, the Planner never observes the preference label at inference; it proposes checks whose truth values are only resolved when the Verifier inspects the image.

The scalar formulation reaches 68.6 Overall, and conditioning it on DeltaRubric checklists improves it to 72.2 Overall and from 61.3 to 64.6 Macro, with gains across all three subcategories. This shows that DeltaRubric generalizes to scalar rewards: the checklist supplies verifiable structure that helps even a pointwise scorer. A pointwise Planner that writes checks about the claims of a single response is a natural direction for further improving scalar rewards.

Table 14: **Comparison with pairwise preference learning and scalar rewards.** VL-RewardBench accuracy under matched backbone (Qwen3-VL-8B Instruct) and training data. Rationale: whether the method outputs an explanation with its decision.

Method	Rationale	Gen.	Hall.	Reas.	Overall	Macro
Qwen3-VL-8B Instruct	Yes	47.0	72.4	43.2	61.3	54.2
DPO (implicit reward)	No	58.0	77.0	53.3	68.2	62.8
Bradley–Terry scalar head	No	49.7	78.8	55.5	68.6	61.3
+ DeltaRubric checklist	No	52.5	82.8	58.5	72.2	64.6
DeltaRubric	Yes	**59.7**	**88.3**	**72.6**	**80.1**	**73.5**

A.10 COMPUTE-MATCHED BASELINES

DeltaRubric runs the model twice per instance, so we examine whether its gains can be attributed to increased inference-time compute. We add two compute-scaled variants of the no-rubric judge. All no-rubric rows in Table 15 use the same RL-trained no-rubric 8B checkpoint and differ only at inference (the two compute-scaled variants are evaluated in a separate inference run of this checkpoint), whereas DeltaRubric uses its own trained Planner-Verifier checkpoint.

- **Verbose CoT.** We append a reasoning instruction to the no-rubric prompt that asks the judge to reason in detail, examine the image systematically, and assess each factual claim in turn.
- **Best-of-3 vote.** We sample three independent verdicts and take the majority vote (Wang et al., 2023b); the arm is charged for all three passes.

We count generated tokens per VL-RewardBench item with the Qwen3-VL tokenizer, excluding prompts; for DeltaRubric, the count includes both the checklist and the Verifier output. As shown in Table 15, DeltaRubric uses fewer tokens than both compute-scaled baselines and only $1.10\times$ those of the no-rubric judge. The generated checklists are concise (typically 2–4 checks, 97 tokens on average), and reasoning over a small set of named checks requires less free-form text than unconstrained reasoning, which reduces the Verifier's generation cost. Despite the smaller budget, DeltaRubric outperforms verbose CoT and best-of-3 by 5.8 and 4.3 Overall points, respectively. These results indicate that the gains of DeltaRubric are not due to more generated tokens, but to the learned rubric-guided planning and verification process.

Table 15: **Compute-matched baselines.** VL-RewardBench accuracy and generated tokens per item (prompt excluded; ratio relative to the no-rubric judge) for Qwen3-VL-8B Instruct.

Method	Tokens	Gen.	Hall.	Reas.	Overall	Macro
No rubric	277 (1.00×)	55.8	86.1	48.3	72.0	63.4
+ Verbose CoT	777 (2.81×)	56.6	86.0	57.1	74.3	66.6
+ Best-of-3 vote	908 (3.28×)	57.2	87.2	59.5	75.8	68.0
DeltaRubric	305 (1.10×)	**59.7**	**88.3**	**72.6**	**80.1**	**73.5**

A.11 EXTERNALLY GENERATED CHECKLISTS

To test whether a more capable but external Planner would outperform our jointly trained one, we use GPT-5.5 (OpenAI, 2026) to generate one checklist per VL-RewardBench item offline, given the image, the question, and both responses, with the same Planner prompt (Appendix A.17) and the same filter (Appendix A.13). The checklists are then executed unchanged by our trained 8B Verifier, so only the source of the checklist changes. As shown in Table 16, RL training improves the resident Planner from 78.8 (frozen) to 80.1 Overall, closing most of the gap to the GPT-5.5 Planner (80.3 Overall, 73.6 Macro). Our Planner achieves this without an external API call, generating 97 tokens per instance on the already deployed model.

Table 16: **Externally generated checklists.** VL-RewardBench accuracy (8B) when only the source of the checklist executed by our trained Verifier changes.

Checklist Source	Overall	Macro
None (Qwen3-VL-8B Instruct base)	61.3	54.2
Frozen 8B Planner (no Planner RL)	78.8	71.3
RL-trained 8B Planner (DeltaRubric)	80.1	73.5
GPT-5.5, offline	80.3	73.6

A.12 CHEAP PROBE VALIDITY

The Planner reward (Eq. 1) is computed with a cheap probe that returns a verdict without extended reasoning, rather than with the full Verifier. To validate this proxy, we hold out 2,000 RLAIF-V

samples disjoint from the 30K training pairs and evaluate both the checklist-conditioned cheap probe and the full Verifier on this set every five RL training steps of the 8B model. Across the 24 evaluation points, the Pearson correlation between the probe accuracy and the full Verifier accuracy is $r = 0.92$, indicating that the probe closely tracks the full Verifier throughout training. Consistently, as the Planner learns to generate better checklists, the Planner probe accuracy (Figure 2c) and the Verifier accuracy (Figures 2a and 2b) improve together rather than diverging.

A.13 Checklist Filtering

Every checklist passes a deterministic filter before it reaches the Verifier, including the sampled and greedy checklists during training and the checklists generated at inference. A checklist is rejected if (1) it is empty or does not consist of 2–4 numbered items, or (2) it contains, case-insensitively, any of the following substrings, grouped into five categories:

- *Candidate identifiers:* "response a", "response b".
- *Verdict markers:* "winner:", "[[a]]", "[[b]]".
- *Comparative claims:* "a is better", "b is better", "a is correct", "b is correct".
- *Preference directives:* "prefer a", "prefer b", "choose a", "choose b".
- *Answer assertions:* "correct answer is", "correct response", "incorrect response", "therefore a", "therefore b".

A rejected checklist is discarded without resampling, and the item is judged with the no-rubric prompt. Because the filter is a fixed format-and-string test rather than a learned judgment, it acts as an external constraint applied at every step, which the model cannot optimize around or relax during training. It ensures that a checklist cannot name a candidate, assert which response is correct, or state a conclusion, so that no verdict-level biases from the planning stage can propagate into verification.

A.14 Faithful Checklist Execution

Since the Verifier is trained only through the correctness of its final verdict, we test whether it faithfully executes the checklist rather than pattern-matching on its presence. On the 8B model, we take the VL-RewardBench items that DeltaRubric answers correctly and use GPT-5.5 to negate a single decisive checklist item, reversing its proposition while leaving every other item unchanged (e.g., "Are the shoes black, not white?" becomes "Are the shoes white, not black?"); the rewriting prompt is given in Appendix A.17. We then rerun the trained Verifier. As shown in Table 17, negating the decisive item lowers Overall accuracy by 15.7 points to 64.4, below the 72.0 no-rubric baseline, and Hallucination accuracy from 88.3 to 76.4. A Verifier that merely pattern-matched on the existence of a checklist would be unaffected by this manipulation. This confirms that the Verifier reads and executes the checks.

Table 17: **Faithful checklist execution.** VL-RewardBench accuracy of the DeltaRubric-trained Qwen3-VL-8B Instruct Verifier when the decisive checklist item is negated.

8B Configuration	Gen.	Hall.	Reas.	Overall	Macro
No rubric	55.8	86.1	48.3	72.0	63.4
DeltaRubric, decisive item negated	48.1	76.4	45.5	64.4	56.7
DeltaRubric, own checklist	**59.7**	**88.3**	**72.6**	**80.1**	**73.5**

A.15 Downstream Policy Training

Policy training. We train Qwen3-VL-8B Instruct as the policy model with GRPO for 120 steps on 17.4K prompts from the RLAIF-V dataset (one prompt per image-question pair). The policy training set is disjoint from the 30K reward model training pairs and is decontaminated against all evaluation benchmarks. We compare two reward models that provide the reward signal, the no-rubric 8B reward model and the DeltaRubric 8B reward model (Table 1), both trained from the same Qwen3-VL-8B base and kept frozen during policy training; all other settings are identical across arms. For each

prompt, the policy samples $G = 8$ free-form responses. Since GRPO normalizes rewards within each group and is invariant to their affine rescaling, a ranking of the group suffices, so the pairwise reward model can be used without adaptation: the responses are ranked by $G - 1$ pairwise comparisons against an anchor response from the same group, with the A/B order randomized to avoid position bias. For DeltaRubric, each comparison consists of a Planner pass followed by a Verifier pass. We use a learning rate of 1×10^{-6}, a global batch size of 128, and a rollout batch size of 256.

Evaluation. We evaluate the resulting policies on AMBER (Wang et al., 2023a), using the accuracy on its discriminative yes/no hallucination questions; MMHal-Bench (Sun et al., 2024), using the mean score (0–6) on its free-form hallucination-oriented questions judged by GPT-5.5 with the benchmark's scoring template; and MMStar (Chen et al., 2024b), using the multiple-choice accuracy to assess general capability. Higher values indicate better performance for all metrics. As shown in Table 6, the policy trained with the DeltaRubric reward model consistently outperforms the one trained with the no-rubric reward model across all three benchmarks, improving AMBER accuracy from 88.7 to 90.9, the MMHal score from 3.56 to 4.14, and MMStar accuracy from 61.2 to 63.7. This suggests that the gains in reward model quality transfer to improved policy optimization and downstream post-training performance.

A.16 Qualitative Examples

We present qualitative examples comparing DeltaRubric against the standard no-rubric baseline. As illustrated in the following cases (Figure 3, 4, 5 and 6), the monolithic baseline frequently suffers from visual hallucinations and flawed reasoning. By contrast, DeltaRubric explicitly enforces rigorous visual verification through its generated checklists, systematically mitigating these errors to ensure an accurate final evaluation.



Question: Describe this image in detail.

Response A (Rejected)

The image captures the hustle and bustle of a busy street, with people walking about and cars driving on the road. A woman stands out in the center, holding a shopping bag and an umbrella to protect her from the rain or sun. Storefronts and shop signs indicate a thriving commercial area. In the background, a bus can be seen, possibly providing public transportation. The group of people adds to the lively atmosphere...

Response B (Chosen)

The image presents a busy street scene with multiple people walking about. A woman is prominently standing toward the center, holding a shopping bag and an umbrella to protect her from the rain. She appears to be coming out of a shopping session and walking down the street.
Several other people are also walking, some holding umbrellas... There are instances of handbags and a cell phone being carried. In the background, a bus can be seen, possibly providing public transportation....

DeltaRubric ✓

Generated Checklist: 1. Is the woman holding a checkered umbrella, not just any umbrella? 2. Is there no visible car traffic on the street, only pedestrians and a bus?
Analysis: 1. The umbrella has a clear checkered pattern. Response A says "an umbrella to protect her from the rain or sun", correct, but doesn't specify checkered. Response B says "an umbrella to protect her from rain or the sun", correct, but doesn't specify checkered. Favors: Neither. 2. No cars are visible; only pedestrians and a bus. Response A says "cars driving on the road", contradicted. Response B says "a bus can be seen", correct. Favors: B.
Justification: Response B is more accurate in describing the absence of cars, Response A incorrectly claims cars are present, which contradicts the image. Therefore, B is better.
Winner: [[B]]

No Rubric ✗

Analysis: Response A claims the image shows a busy street with people walking and cars driving, and that the woman is holding an umbrella to protect from rain or sun. It also mentions storefronts and a bus in the background. Response B implies the woman is "coming out of a shopping session," which is speculative.
Justification: Response A avoids adding unsupported details and correctly describes the scene without fabricating elements like cars, or multiple umbrella-holding pedestrians.
Winner: [[A]]

Figure 3: **Qualitative comparison of evaluation methods.** While the standard no-rubric baseline misses the hallucinated "cars" in Response A, DeltaRubric generates a targeted disagreement checklist. This explicitly enforces visual verification, allowing the model to successfully catch the hallucination and correctly select Response B.

A.17 Prompt Templates

Below, we provide the complete set of prompts used to instruct the model to produce structured outputs. This includes the evaluation prompts for the no-rubric and static-rubric baselines, along with the static rubrics, as well as the DeltaRubric evaluation prompts. Additionally, we detail the Planner prompts used to generate the checklists and the prompts used for the cheap verdicts. In all templates, the image is provided to the model as visual input preceding the text. The static-rubric and DeltaRubric evaluation prompts share the same structure and differ in the rubric slot and in how the Verifier is instructed to use it. Finally, we provide the prompt used to negate a checklist item for the faithful execution test (Appendix A.14).



Question: What are the main objects or subjects in the image? Please describe them in detail.

Response A (Chosen)
The image primarily showcases a person's lower body and parts of their attire. The individual is wearing leggings, shoes, and holding a backpack. There are also elements like stairs in the background.

Response B (Rejected)
The image features a person standing on what appears to be stairs. The individual is holding a backpack in one hand and is dressed in black leggings, white shoes with black laces, and a multi-colored jacket or coat.

DeltaRubric ✓

Generated Checklist: 1. Are the shoes black with white soles and black laces, not white shoes? 2. Is the person standing on stone steps?
Analysis: 1. The shoes are black with white soles and black laces. Response A describes the shoes as "shoes" without specifying color, but the context implies black with white soles. Response B incorrectly states "white shoes with black laces," which contradicts the image. Favors: A. 2. The person is standing on stone steps, and their shadow is cast to the right. Response A correctly notes "stairs in the background." Response B correctly notes "stairs,". Favors: A.
Justification: Response A accurately describes the main subjects (person, leggings, shoes, backpack, stairs) without including incorrect details (e.g., "white shoes" in B). It is concise, accurate, and avoids contradictions.
Winner: [[A]]

No Rubric ✗

Analysis: Response A claims the image primarily shows the person's lower body and parts of their attire, including leggings, shoes, and a backpack, with stairs in the background. Response B claims the person is standing on stairs, holding a backpack, and is dressed in black leggings, white shoes with black laces, and a multi-colored jacket or coat.
Justification: Response B correctly identifies the jacket as multi-colored (maroon and black visible in the image), whereas Response A omits this detail. The shoes are white with black laces, which B correctly notes, while A only vaguely refers to "shoes" without specifying color or laces. B's description is more complete and accurate.
Winner: [[B]]

Figure 4: **Qualitative comparison of evaluation methods.** The standard no-rubric baseline fails to verify fine-grained visual details and incorrectly validates the hallucinated "white shoes" in Response B. In contrast, DeltaRubric generates a targeted checklist that explicitly isolates the conflicting shoe color. By enforcing active visual verification, DeltaRubric successfully catches the hallucination and correctly selects Response A.



Question: What type of food are the people offering the giraffe?

Response A (Rejected)
People are giving a giraffe a section of tree, possibly a tree branch with leaves.

Response B (Chosen)
This person seems to be feeding a giraffe with some kind of dried fruit, which looks yellow in the picture.

DeltaRubric ✓

Generated Checklist: 1. Is the food being offered small, granular, not a large piece of tree or branch? 2. Is the food held in the person's open palm, not attached to a branch or tree limb? 3. Is the giraffe taking the food from the palm, not from a branch or tree trunk? 4. Is there no visible tree branch or foliage being offered; the food is in the person's hand?
Analysis: 1. The food is granular and light-brown, consistent with pellets or dried leaves, not a branch. Favors: B. 2. Food is in open palm, not attached to a branch. Favors: B. 3. Giraffe is taking from palm, not from trunk. Favors: B. 4. No branch or foliage offered, food is in hand. Favors: B.
Justification: All checklist items confirm the food is in the person's hand, granular, and not a branch or foliage.
Winner: [[B]]

No Rubric ✗

Analysis: Response A claims the giraffe is being fed a tree branch, which contradicts the image showing hands offering small, granular food. Response B claims dried fruit, but the food appears granular and not yellow, and no fruit is visible.
Justification: The food in the image is granular and likely pellets or hay, not a tree branch or yellow dried fruit. Both responses misidentify the food, but A is closer to the actual scene since the giraffe is eating from a hand, not a branch.
Winner: [[A]]

Figure 5: **Qualitative comparison of evaluation methods.** The standard no-rubric baseline exhibits severe logical inconsistency: it correctly notes the absence of a tree branch but unaccountably still chooses Response A. In contrast, DeltaRubric generates a targeted checklist that systematically verifies the visual evidence (e.g., granular food in an open palm, not a branch), strictly enforcing logical consistency and correctly select Response B.

No-Rubric Evaluation Prompt

You are a fair judge. Decide which response better answers the question below based on the image.
Question: {question}
Response A: {response_a}
Response B: {response_b}
Compare the two responses and assess which one is better. Then give your overall judgment.
Analysis: <compare both responses>
Justification: <overall reasoning>
Winner: [[A]] or [[B]]



Question: What color is the tissue box on the counter?

Response A (Chosen)
The tissue box on the counter in the picture is green.

DeltaRubric ✓

Generated Checklist: 1. Is there a tissue box visible on the countertop in the lower-left corner of the image? 2. Does the tissue box have a green exterior or cover? 3. Does the tissue box appear to be white or off-white?
Analysis: 1. Yes, a tissue box is visible in the lower-left corner. Favors: Neither. 2. The tissue box has a green exterior (partially visible). Favors: A. 3. Not applicable, the tissue box is not white. Favors: A.
Justification: The tissue box has a green exterior.
Winner: [[A]]

Response B (Rejected)
The tissue box on the counter in the picture is white.

No Rubric ✗

Analysis: Response A claims the tissue box is green, while Response B claims it is white. The image shows a tissue box with a green top and white tissue paper visible, but the box itself appears to be white or off-white.
Justification: The tissue box in the image is clearly white, not green. The green is only on the top or lid, not the box itself. Response B correctly identifies the box as white, while Response A misidentifies it as green.
Winner: [[B]]

Figure 6: **Qualitative comparison of evaluation methods.** The standard no-rubric baseline struggles with fine-grained visual attribute binding, confusing the white tissues with the actual color of the box to incorrectly select Response B. In contrast, DeltaRubric generates a targeted checklist that explicitly isolates and verifies the "green exterior," preventing this attribute confusion and correctly selecting Response A.

Static-Rubric Evaluation Prompt

You are a fair judge. Decide which response better answers the question below based on the image.
Question: {question}
Evaluation Criteria: {rubric}
Response A: {response_a}
Response B: {response_b}
Use the rubric as guidance, not as evidence. If any criterion is irrelevant, too vague, or contradicted by the image/question, ignore that criterion.
For each evaluation criterion, compare the two responses and assess which one is better. Then give your overall judgment.
Analysis:<evaluate criterion by criterion>
Justification: <overall reasoning>
Winner: [[A]] or [[B]]

Static Rubrics

1. Directly answers the question using the information relevant to the image.
2. Makes factual claims that are consistent with the image and avoids unsupported details.
3. Correctly identifies important visual information when it matters for the question.
4. Uses sound reasoning and logical inference where needed.
5. Gives a clear and complete answer.

DeltaRubric Evaluation Prompt

You are a fair judge. Decide which response better answers the question below based on the image.
Question: {question}
Verification Checklist:{checklist}
Response A: {response_a}
Response B: {response_b}
Use the verification checklist as a sequence of checks to execute, not as evidence. If a checklist item is irrelevant, too vague, or contradicted by the image/question, ignore that item. Execute the checklist item by item. For each item, state the evidence and which response it favors. Keep the full answer concise.
Analysis:<work through each checklist item with evidence>
Justification: <one short sentence aggregating the checklist results>
Winner: [[A]] or [[B]]

Planner Prompt

You are preparing an executable verification checklist for judging which of two responses better answers a visual question. Read both responses, identify the decisive disagreements, and write a short checklist that can be executed item by item.
Question: {question}
Response A: {response_a}
Response B: {response_b}
Write a numbered list of 2-4 checks. Rules:
- Each check must describe exactly one concrete fact, relation, or constraint to verify from the image.
- Focus strictly on decisive disagreements or contradictory claims in the responses, not generic advice.
- Keep each check neutral and evidence-seeking.
- Do NOT mention Response A or Response B by name.
- Do NOT say which response is better or correct.
- No preamble, no explanation, only the numbered checks.
Verification Checklist:

No Rubric Cheap Verdict Prompt

You are a fair judge. Decide which response better answers the question below based on the image.
Question: {question}
Response A: {response_a}
Response B: {response_b}
Answer ONLY with [[A]] or [[B]].

Checklist Cheap Verdict Prompt

You are a fair judge. Decide which response better answers the question below based on the image.
Question: {question}
Verification Checklist: {checklist}
Response A: {response_a}
Response B: {response_b}
Use the verification checklist only as a shortlist of checks. Answer ONLY with [[A]] or [[B]].

Checklist Negation Prompt (GPT-5.5)

Below is a numbered checklist of visual checks.
Rewrite it so that exactly ONE item asks the opposite of what it currently asks, and leave every other item untouched. Pick the item that looks most decisive. Keep the numbering, keep the wording style, and keep the item count identical. Do not add commentary.
Checklist: {checklist}
Rewritten checklist:

A.18 AI ASSISTANCE

We used AI assistance to aid and polish writing, including Abstract, Section 1 Introduction and 2 Related Work. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of AI.