

Let VLMs Grade Their Own Thoughts: A Self-Quantification Approach to Reasoning-Aware Reward Modeling

Supplementary Material

1. Datasets

In line with previous studies [1, 3], we evaluate our model on six video understanding benchmarks: VIS-Bench [10], VideoMMMU [5], MMVU [11], MVBench [6], TempCompass [8], and VideoMME [4]. These are divided into two categories: one for assessing high-order reasoning (VIS-Bench, VideoMMMU, MMVU) and another for measuring foundational video understanding (MVBench, TempCompass, VideoMME). For our evaluation, we adhere to specific settings. On the MMVU benchmark, we use only its multiple-choice subset. For VideoMME, we exclude subtitle files to specifically test the model’s ability to process purely visual information. To ensure fairness and reproducibility, we adopt the same decoding configuration as Video-R1. Specifically, we set `top_p` to 0.001, temperature to 0.01, and the maximum number of output tokens to 1024.

2. Implementation Details

Base Model and Setup. Consistent with prior work, we use Qwen2.5-VL-7B as our base model. Our experiments are conducted on a setup of 16 A100 GPUs (1280 GB).

Training and Inference. During training, we set the video resolution to a fixed size of $128 \times 28 \times 28$, and the number of sampled frames per video ranges from 4 to 16. We use a total batch size of 16, which is distributed as 1 sample per device. For each query, we sample 8 candidate responses. The maximum number of RL optimization steps is set to 1000. At inference time, we increase the maximum video resolution to $256 \times 28 \times 28$. We apply different strategies based on the question type: the SCORE strategy is used for multiple-choice questions, while IGSR is applied to all other cases. To ensure a fair comparison and control for the influence of the training data, we use the Video-R1-260k [3] dataset for all training experiments.

Optimization and Hyperparameters. We use the Adam optimizer with a learning rate of $1e-6$, a weight decay of 0.01, and a cosine learning rate schedule. To better align with the semantics of open-ended answers, we replace the ROUGE-1 score with PrefBERT [7]. Furthermore, similar to CARE, we skip the Supervised Fine-Tuning (SFT) stage and directly apply RL to the pre-trained Qwen2.5-VL. We perform Exponential Moving Average (EMA) updates on the reference model with an update interval of 10 and a decay rate of 0.995. No KL penalty is applied during the RL training process. The parameters for our reward strategies are set empirically as follows: for SCORE, we set

$\eta=0.5, \lambda=2.0$; for IGSR, we set $\tau=0.01, \delta=1e-4$, and $\vartheta=0.1$. Further comparisons regarding the confidence evaluation model are provided in Section X.

Code. Our code will be made publicly available upon completion of an internal security and compliance review.

Distribution Details. We employ the Qwen2.5-VL model to sample responses from the Video-R1-260K dataset. For each question, we generate eight candidate responses. The maximum video resolution is set to 128×128 , with a maximum of 16 sampled frames. For images, the resolution is set to 256×256 . The left plot is based on the minimum answer token probability, which corresponds to the ‘minimum probability bottleneck’ targeted by our SCORE method. The right plot, in contrast, shows the statistics for the average token probability of the entire response sequence.

3. Motivation

3.1. Theoretical Motivation

Intrinsic Confidence as a Proxy for Reasoning Quality. Let a VLM be modeled as an autoregressive distribution $\pi_{\text{ref}}(\cdot | q)$ over token sequences conditioned on a video-question pair q . For any generated output $o = (o_{<L}, o_{\geq L})$, where $o_{<L}$ denotes the CoT reasoning path and $o_{<L}$ the answer tokens, the joint probability factorizes as:

$$\pi_{\text{ref}}(o | q) = \pi_{\text{ref}}(o_{<L} | q) \cdot \pi_{\text{ref}}(o_{\geq L} | q, o_{<L}) \quad (1)$$

Define the answer confidence as the geometric mean of the per-answer-token predictive probabilities as estimated by the frozen reference model:

$$c(o_{\geq L} | q, o_{<L}) := \left(\prod_{t=L}^T \pi_{\text{ref}}(o_t | q, o_{<t}) \right)^{1/(T-L+1)}. \quad (2)$$

Let \mathcal{O}^* be the set of outputs whose reasoning paths are logically valid and entail the ground-truth answer a^* . Our core hypothesis is that a well-calibrated reference model, when conditioned on a sound reasoning path, will be more confident in the correct answer than when conditioned on a flawed one.

Core Assumption. For any $o \in \mathcal{O}^*$ (with reasoning $o_{<L}$ and answer a^*), the contextual prefix $o_{<L}$ provides sufficient and consistent evidence for the correct answer, enabling the reference model to assign high probability to each token in $a^* = o_{>L}$. Formally, for all $t \in [L, |o|]$,

$$\pi_{\text{ref}}(o_t | q, o_{<t}) \geq \pi_{\text{ref}}(o'_t | q, o'_{<t}), \quad \forall o' \notin \mathcal{O}^*, \quad (3)$$

System: You are a helpful assistant.

User:

Your task is to answer the question by selecting the correct option based strictly on the provided reasoning process.

I will supply the question, its answer options, and a detailed reasoning process. You must choose the most appropriate answer from the provided options, ensuring consistency with the reasoning process. Please provide only the single option letter (e.g., A, B, C, D, etc.) within the <answer> </answer> tags.

Question: {Question}

Options: {Options}

Reasoning Process: {Process}

Query (multiple choice):

System: You are a helpful assistant.

User:

Your task is to answer the question by selecting the correct option based strictly on the provided reasoning process.

I will supply the question, its answer options, and a detailed reasoning process. You must choose the most appropriate answer from the provided options, ensuring consistency with the reasoning process. Please provide only the single option letter (e.g., A, B, C, D, etc.) within the <answer> </answer> tags.

Question: Measuring from the closest point of each object, which of these objects (door, headphones, monitor, whiteboard) is the closest to the telephone?

Options: [A. door, B. headphones, C. monitor, D. whiteboard]

Reasoning Process: <think>Let me think through this step by step. The telephone is placed on the desk, and we need to determine which object is closest to it. The door is further away, the headphones are on the desk but not directly next to the telephone, the monitor is also on the desk but not immediately adjacent to the telephone. The whiteboard is on the wall, which is further from the telephone. Let's break this down: the telephone is on the desk, and the monitor is right next to it. This makes the monitor the closest object to the telephone. Oh, I see it now. The answer is C.</think>

DeepSeek V3:

Responses: <answer>C</answer>

Figure 1. **Prompt Template for Reasoning Path Evaluation.** Top: The prompt template used to query the text-only LLMs. It includes three placeholders for the Question, the Options, and the Process. Bottom: An instantiated example of the template, populated with an output from our Video-RAISE model on a sample from the VIS-Bench benchmark.

Method	Video Reasoning Benchmark						Video General Benchmark					
	VSI-Bench		VideoMMMU		MMVU		MVBench		TempCompass		VideoMME	
	Answer	Match	Answer	Match	Answer	Match	Answer	Match	Answer	Match	Answer	Match
Qwen2.5-VL	10.0	17.2	46.9	46.7	54.2	55.5	24.0	35.9	63.4	87.8	36.8	25.0
Qwen2.5-VL-SFT	16.3	41.2	47.4	87.8	61.9	52.3	52.2	86.6	65.4	93.5	49.6	90.5
Qwen2.5-VL-GRPO	18.5	41.4	40.4	34.0	45.7	47.2	29.1	26.9	47.3	43.7	32.9	37.3
Ours: Video-RAISE	30.3	84.9	55.3	87.9	66.7	94.9	63.5	95.5	73.5	95.9	56.1	93.4

Table 1. **Analysis of Reasoning Paths.** The prompt templates and examples used for this analysis are shown in Figure 7. Match indicates the consistency (exact match) between the LLM-generated answer and the original VLMs answer, while Answer measures the correctness of the LLM-generated answer. Our Video-RAISE demonstrates a significant lead on both metrics.

where the inequality holds because any $o' \notin \mathcal{O}^*$ contains a reasoning flaw, such as factual inconsistency, logical gap, or misalignment with visual evidence, that weakens the predictive support for the correct answer tokens. Since the geometric mean is a strictly increasing function of its arguments

over $(0, 1]$, it follows directly that:

$$c(a^* | q, o_{<L}) > c(a^* | q, o'_{<L}), \forall o \in \mathcal{O}^*, o' \notin \mathcal{O}^*. \quad (4)$$

Thus, the intrinsic confidence $c(\cdot)$ serves as a faithful scalar indicator of reasoning quality: higher confidence implies a reasoning path that is more likely to be logically valid and

Method	Frame	Video Reasoning Benchmark			Video General Benchmark		
		VSI-Bench	VideoMMMU	MMVU	MVBench	TempCompass	VideoMME
Qwen2.5-VL-SFT-GRPO	16	34.4	50.1	63.2	64.0	73.2	56.9
Qwen2.5-VL-GRPO	16	33.6	47.7	65.3	63.8	73.9	57.0
Video-RAISE-Policy-Instruct	16	35.6	52.4	67.2	64.4	74.8	58.4
Video-RAISE-Policy-SFT	16	36.5	52.1	64.5	63.5	72.6	56.5
Video-RAISE-Reference-Instruct	16	34.8	52.8	66.4	64.3	74.7	57.4
Video-RAISE-Reference-SFT	16	36.0	49.9	61.4	63.5	72.1	57.3
Video-RAISE-Policy-KL	16	33.5	52.3	66.4	64.1	73.7	55.9
Video-RAISE-Reference-KL	16	31.9	52.2	65.3	63.1	73.8	55.3
Video-RAISE-Policy-Instruct	32	37.0	52.3	65.9	66.2	75.3	60.4
Video-RAISE-Reference-Instruct	32	36.6	53.0	65.9	65.9	75.1	60.7
Video-RAISE-Policy-Instruct	64	39.2	53.0	67.0	66.3	75.2	61.0
Video-RAISE-Reference-Instruct	64	38.7	53.6	66.9	65.9	75.0	62.6

Table 2. **Ablation Study on Training Configurations.** SFT refers to the model fine-tuned on Video-R1-260k [3]. The Policy and Reference settings indicate whether the reward is calculated using probabilities from the policy or reference model, respectively. KL indicates the application of a KL divergence penalty. The results show that calculating the reward based on either the policy or reference model yields strong performance, with each approach having its own distinct advantages.

Method	#Parm	Video Reasoning Benchmark			Video General Benchmark		
		VSI-Bench	VideoMMMU	MMVU	MVBench	TempCompass	VideoMME
	0.1	34.7	52.6	66.1	64.7	74.7	57.9
Score Combine:	0.3	35.8	51.4	65.8	64.1	74.6	57.4
$b \cdot \pi_{\text{ref}}(o_{i,j} q, o_{i,<j}) +$	0.5	35.2	50.7	67.4	64.9	74.8	56.9
$(1 - b) \cdot \pi_{\theta}(o_{i,j} q, o_{i,<j})$	0.7	35.0	49.5	66.1	63.8	74.4	57.4
	0.9	33.8	49.8	65.6	64.2	74.0	57.2
	0.1	34.5	50.6	66.7	64.3	74.2	57.0
Reward Combine:	0.3	34.5	51.8	65.8	63.1	74.4	56.8
$r_{(\text{ref})}^b \cdot r_{(\theta)}^{1-b}$	0.5	34.5	51.9	66.7	63.7	73.0	57.6
	0.7	35.4	52.8	66.6	63.7	74.3	57.6
	0.9	33.5	52.3	66.4	64.1	73.7	55.9

Table 3. **Analysis of the Trade-off between the Policy and Reference Models.** This table compares two combination strategies: Score Combine, which merges the models at the probability (score) level, and Reward Combine, which merges them at the final reward level. The hyperparameter b balances the contribution of each model, and its value is shown in the #parm column.

supportive of the correct answer. Consequently, optimizing the policy to maximize expected confidence induces alignment with high-quality reasoning without external supervision:

$$\max_{\theta} \mathbb{E}_{o \sim \pi_{\theta}(\cdot|q)} [c(o_{\geq L} | q, o_{<L})]. \quad (5)$$

4. Analysis of Reasoning Paths

To evaluate the effectiveness of the reasoning paths generated by our Video-RAISE, we conducted a comparative experiment. For each benchmark, we employed string matching to extract the "think" portion—the content enclosed within the `<think>...</think>` tags—that the model gen-

erates before producing the final answer. This extracted content was then combined with the original question and options and provided as input to a text-only LLMs. The LLMs was then prompted to answer the original question without access to the visual information. Subsequently, we compared the answers generated by the LLMs with those from the original VLMs to assess for consistency and correctness. Our rationale is that if a model’s reasoning path is effective and logical, a text-only LLMs should be able to extract the key information from it to arrive at the correct answer, even without visual input. The prompt template used for the LLM is detailed in Figure 1, and results are presented in Table 1.

Consistency. The baseline model, Qwen2.5-VL-Instruct, shows strong consistency on TempCompass, achieving a matching rate of 87.8%. However, its performance is poorer on other benchmarks such as VideoMMU and VideoMME. We attribute this to the base model’s limited ability to adhere to the specified format. Through case analysis, we found that in a significant portion of its outputs, Qwen2.5-VL-Instruct failed to correctly place the reasoning process within the designated `</think>` tags. In contrast, the SFT model, which distills the reasoning capabilities of more powerful models, exhibits a stronger consistency between its reasoning paths and answers. It achieves over 90% matching accuracy on the TempCompass and VideoMME datasets. However, as noted in previous research [9], SFT models tend to perform poorly in scenarios with data dissimilar to their training set, known as out-of-distribution (OOD) data. On the MMVU benchmark, its matching accuracy was only 52.3%. As for the GRPO method, because it does not impose explicit constraints (e.g., rewards) on the reasoning process, it performs poorly across most benchmarks. It shows only a slight improvement on the VSI-Bench and VideoMME benchmarks, with performance declining on the others. Our VideoRAISE, however, discards external constraints during its RL training. Instead, it utilizes the probability of the answer sequence as a reward signal to encourage the VLM to generate reasoning paths that have a higher likelihood of leading to the correct final answer. This approach yields exceptional consistency between the reasoning path and the answer, achieving a matching rate of nearly 90% across all benchmarks.

Correctness. VideoRAISE enables the LLM to achieve comparable or even superior accuracy. On the Video-MMMU benchmark, the LLM using VideoRAISE’s reasoning path achieved an accuracy of 55.3%, surpassing the performance of the original VLMs that had access to visual information. On other benchmarks, it achieves similar results; for instance, on TempCompass, it reached a correctness of 73.5%, only 1.2 points lower than the fully-equipped VLM. In contrast, other methods exhibit a significant performance degradation. For example, while Qwen2.5-VL-SFT achieved a high consistency rate of 93.5% on TempCompass, its corresponding correctness was only 65.4%. The performance drop is even more pronounced on out-of-distribution datasets; on the MMVU benchmark, the accuracy of the LLMs using Qwen2.5-VL-SFT’s reasoning path dropped by 9 percentage points. These findings validate the effectiveness of our proposed reasoning paths.

5. Training Configuration

To identify the optimal training configuration, we present a detailed comparison of different settings in Table 2. Our analysis is structured around two key questions: (1) whether

a preliminary Supervised Fine-Tuning (SFT) stage is beneficial, and (2) whether it is better to calculate rewards using the current policy model or the more stable reference model.

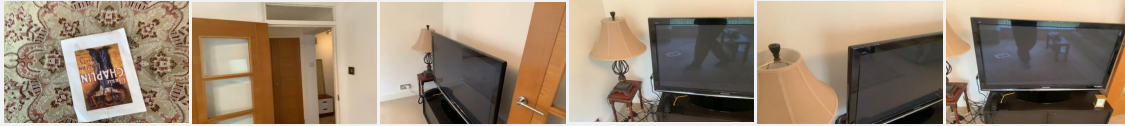
Impact of the Supervised Fine-Tuning (SFT) Stage. Some prior work [2] suggests that using a high-quality dataset for a warm-up SFT stage before RL can be beneficial. We first test this hypothesis. The Qwen2.5-VL-SFT-GRPO model achieves an accuracy of 50.1% on VideoMMU, outperforming its non-SFT counterpart (Qwen2.5-VL-GRPO) by 2.4 points. However, the opposite is true on the MMVU benchmark, where Qwen2.5-VL-GRPO (65.3%) surpasses the SFT-based version by 2.1 points. On other datasets, the two methods show no significant difference and each has its own advantages. Given these mixed results, we conduct separate analyses for models based on the original instruction-tuned version and those that first undergo SFT. Our findings consistently show that starting RL directly from the instruction-tuned model is superior. When we apply our VideoRAISE method to SFT-based models, performance degrades across the board. For example, VideoRAISE-Policy-SFT sees its accuracy drop from 67.2% to 64.5% on MMVU and from 58.4% to 56.5% on VideoMME. The degradation is even more pronounced for the reference model variant, which drops from 66.4% to 61.4% on MMVU and falls to 49.9% on VideoMMM. While SFT can teach a model to follow formats and distill reasoning from larger models, our results suggest it disrupts the general knowledge acquired during large-scale pre-training, thereby limiting the model’s exploration capabilities during RL.

Recommendation: We recommend using the instruction-tuned version (without Supervised Fine-Tuning) as the base model for our reward calculation method.

Policy vs. Reference Model for Reward Calculation. We then compare two approaches for calculating the reward: using the current policy model, which provides a real-time evaluation of the model’s confidence, versus using the reference model (updated via EMA), which provides a more stable probability estimate. At a 16-frame setting, we find that both approaches yield strong performance, each with distinct advantages. VideoRAISE-Policy-Instruct achieves 67.2% accuracy on MMVU, outperforming the reference-based version (66.4%). Conversely, the reference-based model excels on VideoMMM, reaching 52.8%, which not only surpasses the policy-based model but also exceeds the previous state-of-the-art method, VideoR1, by a margin of +3.0 points. As we increase the number of input frames to 32 and 64, the performance gap between the two methods narrows, and in some cases, reverses. At 32 frames: The policy model’s advantage on MMVU is erased, with both methods achieving an identical accuracy of 65.9%. On VSI-Bench, the policy model’s lead shrinks from 0.8

How many sofa(s) are in this room?

Answer: 2



Qwen2.5-VL-Instruct: <think> Let me think...wait, hmm, oh, I see, let's break it down.</think> <answer>1</answer>

Qwen2.5-VL-SFT: <think>Let me think about this carefully. The image shows a room with various pieces of furniture. I see a large beige sofa with a patterned rug underneath it. There's also a white bench with a cushion, but that's not a sofa. Wait, is there another sofa? Hmm, I don't see any other furniture that could be classified as a sofa. Oh, I see, there's a small table with a mirror, but that's not a sofa either. Let's break it down: the beige sofa is the only piece of furniture that fits the description of a sofa. I'm confident in my observation.</think> <answer>1</answer>

Qwen2.5-VL-GRPO: <think>Hmm, let me think through this carefully. The room contains one beige sofa with a plaid throw pillow, which is clearly visible. Let's break it down; there are no other sofas in sight. Therefore, my final answer is 1.</think> <answer>1</answer>

Video-RAISE: <think>Let me think about this for a moment. Hmm, let's break this down. We see a beige sofa with a plaid pillow on it, and another beige sofa with a different pillow. So, there are two sofas in this room. Oh, I see it now. Answer: 2.</think> <answer>2</answer>

You are a robot beginning at the paper shredder box facing the opened door. You want to navigate to the green bucket. You will perform the following actions (Note: for each [please fill in], choose either 'turn back,' 'turn left,' or 'turn right.'): 1. [please fill in] 2. Go forward until the green bucket. You have reached the final destination. Answer: B



A. Turn Right

B. Turn Left

C. Turn Back

Qwen2.5-VL-Instruct: I start at the paper shredder box, which is facing the opened door. To reach the green bucket, I need to move in a direction that will take me towards it. The green bucket is located to the right of the paper shredder box. Therefore, I need to turn right to move towards the green bucket. <answer>A</answer>

Qwen2.5-VL-SFT: <think>Let me think about this carefully. The goal is to navigate from the paper shredder box towards the green bucket. First, I need to identify the location of the paper shredder box and the green bucket in the image. The paper shredder box is on the left side of the image, and the green bucket is on the right side. To get from the paper shredder box to the green bucket, I need to move in a straight line towards the right. let's break it down further. If I start from the paper shredder box, I need to turn right to move towards the green bucket. This is because the green bucket is on the right side of the image. The correct action is to turn right to move towards the green bucket. I don't need to turn back or turn left because those actions would take me away from the green bucket. Let's verify this. If I turn right, I will move in the direction of the green bucket, which is the correct path to take.</think> <answer>A</answer>

Qwen2.5-VL-GRPO: <think> Let me think...wait, hmm, oh, I see, let's break it down.</think> <answer>A</answer>

Video-RAISE: <think>Let me think through this for a moment. We start at the paper shredder box, facing the opened door. To reach the green bucket, I need to move in a direction that will take me towards it. Given the layout, it seems logical to turn left and then proceed forward until I reach the green bucket. Let's break this down: turning left will align me with the path to the bucket, and going forward will get me there. Oh, I see it now. The answer is B.</think> <answer>B</answer>

Figure 2. Case Study. Top: An illustrative example of a counting task. Bottom: The results for a multiple-choice task.

to 0.4 points. More significantly, on VideoMME, the reference model surpasses the policy model, turning the policy model’s 1.0-point advantage into a 0.3-point deficit. At 64 frames: This trend becomes more pronounced. On VideoMME, the reference model’s lead widens to 1.6 points (62.6% vs. 61.0%). Despite this convergence, each method retains its own strengths even at 64 frames, with Video-RAISE-Policy-Instruct still holding an advantage on VSI-Bench, MMVU, MVBench, and TempCompass.

Recommendation: Both the policy and reference models are effective sources for reward calculation. For general use, Video-RAISE-Policy-Instruct can be considered a strong and reliable option.

6. Policy and Reference Combine

In our framework, the policy model reflects the most current parameters, providing a real-time estimate for reward calculation. In contrast, the reference model, updated via EMA, is more stable and provides an averaged estimate. As shown in Table 2, both models achieve comparable performance individually. This motivates us to explore whether a balance between the real-time and stable estimators can yield further improvements. Score-Level Combination. We first investigate a score-level combination. In this approach, we combine the answer probabilities from both the policy and reference models directly using a hyperparameter b to weigh their contributions (Table 3, top). However, this method yields only marginal benefits on specific benchmarks. For instance, on MMVU, a b of 0.5 achieves 67.4% accuracy, a minor improvement of 0.2 points over the best individual model score. Reward-Level Combination. We then propose a reward-level combination. This method combines the final rewards from each model using a geometric weighted average, which places greater emphasis on smaller values (Table 3, bottom). This proves beneficial for video reasoning benchmarks, achieving a new best of 52.8% on VideoMMU while maintaining comparable performance on VIS-Bench and MMVU. However, this combination is detrimental to performance on general-purpose benchmarks, where it consistently underperforms the individual models. Based on these findings, we recommend using either the policy or reference model individually for general applications. Alternatively, the reward-level combination can be adopted when strong reasoning performance is the primary requirement.

7. Case Study

In Figure 2, we present two case studies to qualitatively analyze our model’s performance in reasoning process. The first is an open-ended counting task, and the second is a multiple-choice question that requires spatial reasoning.

Open-Ended Counting Task. In this task, the model

Algorithm 1 Sequential Confidence Rigorous Evaluation (SCORE)

- 1: **Input:**
Video-question pair (v, q) ,
Policy model π_{policy} ,
Reference model π_{ref} ,
Number of candidates G ,
Hyperparameters $\eta > 0, \lambda > 0$.
 - 2: **Output:** A set of rewards $\{\tilde{r}_i\}_{i=1}^G$.
 - 3:
 - 4: Sample G candidate sequences $\{o_i\}_{i=1}^G \sim \pi_{\text{policy}}(\cdot | q)$. \triangleright Generate G candidate answers using the policy model.
 - 5: **for** $i = 1$ to G **do**
 - 6: **Step 1: Compute token-wise probabilities.**
 $\bar{p}_{i,j} = \pi_{\text{ref}}(o_{i,j} | q, o_{i,<j})$.
 - 7: **Step 2: Extract answer probabilities.**
 - 8: $p_i = \text{Match}(\bar{p}_i, \langle \text{answer} \rangle \dots \langle / \text{answer} \rangle)$. \triangleright
 Locate and extract the probability of the answer part.
 - 9: **Step 3: Calculate weighted product reward.**
 $\bar{r}_i = \prod_{j=1}^{|p_i|} p_{i,j}^{w_j}$, where the weight is $w_j = e^{-\eta j}$.
 - 10: **Step 4: Compute variance penalty.**
 $\sigma_i^2 = \frac{1}{|p_i|} \sum_{j=1}^{|p_i|} (p_{i,j} - \mu_i)^2, \mu_i = \frac{1}{|p_i|} \sum_{j=1}^{|p_i|} p_{i,j}$.
 - 11: **Step 5: Apply variance penalty to encourage consistent confidence.**
 $\tilde{r}_i = \bar{r}_i \cdot e^{-\lambda \sigma_i^2}$.
 - 12: **end for**
 - 13:
 - 14: **return** $\{\tilde{r}_i\}_{i=1}^G$.
-

must count the number of sofas in a video and generate the answer directly without options. We compare the reasoning paths of different models. Qwen2.5-VL-Instruct: The base instruction-tuned model fails to perform genuine reasoning. It merely outputs a few text fragments that mimic the format of human-like thinking as defined in the prompt. Qwen2.5-VL-SFT: The model fine-tuned with SFT generates a longer reasoning path. However, it exhibits a "first-find" bias: after identifying the most prominent sofa, it neglects to search for others and prematurely concludes with a count of one, leading to an incorrect answer. Qwen2.5-VL-GRPO: Similarly, the rule-based RL model identifies the largest sofa but then incorrectly assumes there are no others, failing to complete the task. Video-RAISE: In contrast, our model demonstrates a more robust and methodical process. It first enumerates the specific locations of all detected sofas. Based on this complete set of evidence, it makes a correct prediction of two.

Multiple-Choice Spatial Reasoning Task. This task requires the VLMs not only to identify all objects in the video but also to understand the spatial position of the camera operator to make a decision about the next action.

Algorithm 2 Intra-group Score Re-ranking (IGSR)

1: **Input:**
A set of G candidate sequences $\{o_i\}_{i=1}^G$,
A reference answer t_{ref} ,
A reference model π_{ref} ,
Hyperparameters $\tau, \delta, \partial, \epsilon$.

2:

3: **Output:** A re-ranked set of rewards $\{\tilde{r}_i\}_{i=1}^G$.

4: **Phase 1: Scoring and Grouping.**

5: **for** $i = 1$ to G **do**

6: **Step 1: Decode candidate to text.**
 $t_i \leftarrow \text{Decode}(o_i)$.

7: **Step 2: Compute semantic accuracy score.**
 $s_i \leftarrow r(t_i, t_{\text{ref}})$. \triangleright E.g., ROUGE-L similarity to the reference text.

8: **Step 3: Extract answer token probabilities.**
 $p_i \leftarrow \text{Match}(\pi_{\text{ref}}, o_i)$.

9: **Step 4: Compute confidence score.**
 $e_i \leftarrow \frac{1}{|p_i|} \sum_{j=1}^{|p_i|} -\log(p_{i,j} + \delta)$.

10: **end for**

11: **Step 5: Cluster candidates into groups.**
 Assign candidate to group $g^{\{k\}}$ based on interval τ :
 $g^{\{k\}} = \{i : s_i \in [k\tau, (k+1)\tau)\}$.

12: **Step 6: Compute representative rewards.**

13: **for** $k = 1$ to M (number of groups) **do**
 $r_{(m)}^{\{k\}} \leftarrow \text{Median}(\{s_i \mid i \in g^{\{k\}}\})$.

14: **end for**

15: **Phase 2: Intra-group Re-ranking.**

16: **for** each group $g^{\{k\}}$ in $\{g^{\{k\}}\}_{k=1}^M$ **do**

17: **Step 1: Compute reward gap to the next group.**

$$\Delta r \leftarrow \begin{cases} r_{(m)}^{\{k+1\}} - r_{(m)}^{\{k\}} & \text{if } k < M, \\ 1.0 - r_{(m)}^{\{k\}} & \text{if } k = M. \end{cases}$$

18: **Step 2: Compute max and min confidence scores.**
 $e_{\text{max}} \leftarrow \max(\{e_i \mid i \in g^{\{k\}}\})$,
 $e_{\text{min}} \leftarrow \min(\{e_i \mid i \in g^{\{k\}}\})$.

19: **for** each candidate i in group $g^{\{k\}}$ **do**

20: **Step 3: Compute max potential bonus.**
 $b_{\text{max}} \leftarrow \Delta r - \partial \left(1 - \frac{1}{|g^{\{k\}}|+1}\right)$.

21: **Step 4: Compute relative confidence.**
 $c_{\text{rel}} \leftarrow \frac{e_{\text{max}} - e_i}{e_{\text{max}} - e_{\text{min}} + \epsilon}$.

22: ϵ to avoid division by zero.

23: **Step 5: Compute adjustment bonus.**
 $a_i \leftarrow b_{\text{max}} \cdot c_{\text{rel}}$.

24: **Step 6: Compute final re-ranked reward.**

$$\tilde{r}_i \leftarrow \sqrt{r_{(m)}^{\{k\}} \cdot \left(r_{(m)}^{\{k\}} + a_i \cdot \mathbf{1}(a_i \geq \tau)\right)}$$
.

25: $\mathbf{1}(\cdot)$ filters out negligible adjustments.

26: **end for**

27: **end for**

28: **return** $\{\tilde{r}_i\}_{i=1}^G$.

Qwen2.5-VL-Instruct & Qwen2.5-VL-SFT: Both of these models correctly identify objects in the video but fail to adopt the camera operator’s perspective. As a result, they lose track of the spatial context and choose the wrong action. Qwen2.5-VL-GRPO: Similar to its behavior in the counting task, this model fails to generate a meaningful reasoning path. It instead only mimics the stylistic format of human thought from the prompt. Video-RAISE (Ours): Our model, however, correctly reasons from the camera operator’s perspective to deduce the correct action (“move left”). This step-by-step, logical deduction is a direct result of our method’s objective to maximize probability of correct answer tokens.

These qualitative examples provide strong evidence for our core hypothesis: leveraging a VLM’s own internal evaluation of its reasoning paths is a valid and effective approach for end-to-end model training.

8. Pseudocode

References

- [1] Yi Chen, Yuying Ge, Rui Wang, Yixiao Ge, Junhao Cheng, Ying Shan, and Xihui Liu. Grpo-care: Consistency-aware reinforcement learning for multimodal reasoning, 2025. [1](#)
- [2] DeepSeek-AI. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *CoRR*, abs/2501.12948, 2025. [4](#)
- [3] Kaituo Feng, Kaixiong Gong, Bohao Li, Zonghao Guo, Yibing Wang, Tianshuo Peng, Benyou Wang, and Xiangyu Yue. Video-r1: Reinforcing video reasoning in mllms. *CoRR*, abs/2503.21776, 2025. [1](#), [3](#)
- [4] Chaoyou Fu, Yuhan Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, Peixian Chen, Yanwei Li, Shaohui Lin, Sirui Zhao, Ke Li, Tong Xu, Xiawu Zheng, Enhong Chen, Caifeng Shan, Ran He, and Xing Sun. Video-mme: The first-ever comprehensive evaluation benchmark of multimodal llms in video analysis. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2025, Nashville, TN, USA, June 11-15, 2025*, pages 24108–24118. Computer Vision Foundation / IEEE, 2025. [1](#)
- [5] Kairui Hu, Penghao Wu, Fanyi Pu, Wang Xiao, Yuanhan Zhang, Xiang Yue, Bo Li, and Ziwei Liu. Video-mmmu: Evaluating knowledge acquisition from multi-discipline professional videos. *CoRR*, abs/2501.13826, 2025. [1](#)
- [6] Kunchang Li, Yali Wang, Yinan He, Yizhuo Li, Yi Wang, Yi Liu, Zun Wang, Jilan Xu, Guo Chen, Ping Lou, Limin Wang, and Yu Qiao. Mvbench: A comprehensive multimodal video understanding benchmark. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pages 22195–22206. IEEE, 2024. [1](#)
- [7] Zongxia Li, Yapei Chang, Yuhang Zhou, Xiyang Wu, Zichao Liang, Yoo Yeon Sung, and Jordan Lee Boyd-Graber. Semantically-aware rewards for open-ended r1 training in free-form generation, 2025. [1](#)
- [8] Yuanxin Liu, Shicheng Li, Yi Liu, Yuxiang Wang, Shuhuai Ren, Lei Li, Sishuo Chen, Xu Sun, and Lu Hou. Tempcompass: Do video llms really understand videos? In *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024*, pages 8731–8772. Association for Computational Linguistics, 2024. [1](#)
- [9] Graham Neubig, Seungone Kim, Minxin Du, Xiang Yue, Tuney Zheng, Radha Poovendran, Yuetai Li, Xiaoyu Xu, and Maggie Huan. Does math reasoning improve general llm capabilities? understanding transferability of llm reasoning, 2025. [4](#)
- [10] Jihan Yang, Shusheng Yang, Anjali W. Gupta, Rilyn Han, Li Fei-Fei, and Saining Xie. Thinking in space: How multimodal large language models see, remember, and recall spaces. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2025, Nashville, TN, USA, June 11-15, 2025*, pages 10632–10643. Computer Vision Foundation / IEEE, 2025. [1](#)
- [11] Yilun Zhao, Haowei Zhang, Lujing Xie, Tongyan Hu, Guo Gan, Yitao Long, Zhiyuan Hu, Weiyuan Chen, Chuhan Li, Zhijian Xu, Chengye Wang, Ziyao Shangguan, Zhenwen Liang, Yixin Liu, Chen Zhao, and Arman Cohan. MMVU: measuring expert-level multi-discipline video understanding. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2025, Nashville, TN, USA, June 11-15, 2025*, pages 8475–8489. Computer Vision Foundation / IEEE, 2025. [1](#)