

# PDCR: Perception-Decomposed Confidence Reward for Vision-Language Reasoning

Hee Suk Yoon<sup>1\*†</sup> Eunseop Yoon<sup>1\*</sup> Ji Woo Hong<sup>1</sup> SooHwan Eom<sup>1</sup> Gwanhyeong Koo<sup>1</sup>  
 Mark Hasegawa-Johnson<sup>2</sup> Qi Dai<sup>3</sup> Chong Luo<sup>3‡</sup> Chang D. Yoo<sup>1‡</sup>

<sup>1</sup>Korea Advanced Institute of Science and Technology (KAIST)

<sup>2</sup>University of Illinois at Urbana-Champaign (UIUC) <sup>3</sup>Microsoft Research Asia (MSRA)

{hskyoon, esyo97, cd.yoo}@kaist.ac.kr, {jhasegaw}@illinois.edu, {qid, clou}@microsoft.com

## Abstract

Reinforcement Learning with Verifiable Rewards (RLVR) traditionally relies on a sparse, outcome-based signal. Recent work shows that providing a fine-grained, model-intrinsic signal—rewarding the confidence growth in the ground-truth answer—effectively improves language reasoning training by providing step-level guidance without costly external models. While effective for unimodal text, we find that naively applying this global reward to vision-language (V-L) reasoning is a suboptimal strategy, as the task is a heterogeneous mix of sparse visual perception and dense textual reasoning. This global normalization creates mixture-induced signal degradation, where the training signal for visual steps is statistically distorted by the predominant textual steps. We propose **Perception-Decomposed Confidence Reward (PDCR)**, a framework that solves this by aligning the reward structure with the task’s heterogeneous nature. PDCR first performs an unsupervised skill decomposition, introducing a model-internal Visual Dependence Score to quantify visual reliance and applying a clustering algorithm to separate perception and reasoning steps. Based on this, PDCR computes a decomposed advantage by normalizing confidence gains within each skill cluster. This intra-cluster normalization provides a stable, correctly-scaled signal for both perception and reasoning. We demonstrate that PDCR outperforms the naive, global-reward formulation and sparse-reward baselines on key V-L reasoning benchmarks. The code is publicly accessible at <https://github.com/hee-suk-yoon/PDCR>.

## 1. Introduction

Pre-trained vision-language models (VLMs) exhibit strong performance on complex, multi-step reasoning tasks [1, 2,

\*Equal Contribution.

†This work was done while Hee Suk Yoon was an intern at MSRA.

‡Corresponding Authors.

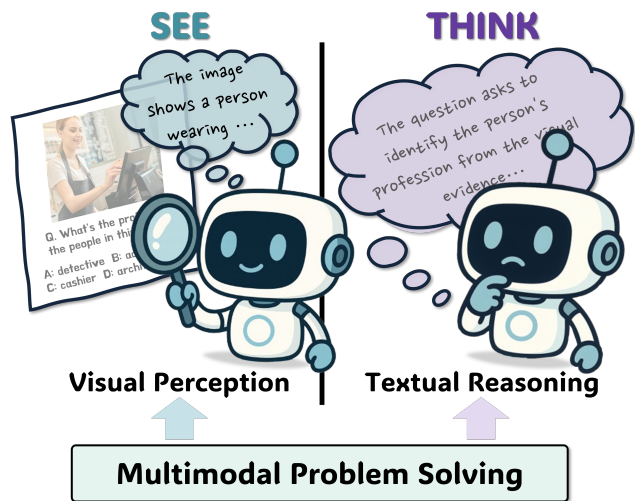


Figure 1. Multimodal reasoning mixes two distinct behaviors: seeing (visual perception, extracting evidence from the image) and thinking (textual reasoning over that evidence). Our work argues that these heterogeneous skills must be rewarded independently, as a naive, global reward signal fails to properly assign credit to each.

16, 29, 39]. Reinforcement Learning with Verifiable Rewards (RLVR) has emerged as a leading approach for further improving these capabilities, using a programmatically checkable terminal metric (e.g., correctness of the final answer) as the reward [9, 11, 28, 35, 38]. While effective, the standard RLVR formulation supplies a sparse terminal accuracy signal, offering no guidance for intermediate steps and thus creating a credit assignment problem.

To address this, recent work has explored fine-grained, step-level supervision. One prominent approach is training an external Process Reward Model (PRM) [3, 21, 23, 31, 41]. However, PRMs are often costly to train, data-hungry, and prone to misalignment. In contrast, a compelling, model-intrinsic alternative from the language domain is to reward the ground-truth confidence growth [42]. This method pro-

vides step-level guidance using the model’s own probabilities of the ground-truth answer, avoiding the need for an external reward model to evaluate each reasoning step.

While effective for unimodal text, we find that naively applying this global reward to **vision-language (V-L) reasoning** is a suboptimal strategy. As illustrated in Figure 1, the core challenge is that V-L reasoning is not a homogeneous process; it is a **heterogeneous mixture of two distinct, high-level skills**: (i) sparse, but critical, **visual perception** steps and (ii) dense, predominant **textual reasoning** steps. This statistical imbalance, as we show in Section 4, leads directly to mixture-induced signal degradation: the reward signal for critical visual steps is statistically distorted by the predominant textual steps.

To this end, we propose Perception-Decomposed Confidence Reward (PDCR), a framework that solves this by aligning the reward structure with the task’s heterogeneous nature. As illustrated in Figure 4, PDCR operates in two stages:

1. **Unsupervised Skill Decomposition**: We first introduce a Visual Dependence Score, a model-internal signal that quantifies a step’s reliance on the visual input. We then apply an optimal clustering algorithm (Otsu’s method) to automatically separate steps into **visual perception** and **textual reasoning** groups.
2. **Perception-Decomposed Advantage**: Based on this, PDCR computes a decomposed advantage by normalizing confidence gains *within* each respective skill cluster, ensuring a stable, intra-cluster comparison.

This decomposed approach, when paired with the standard terminal reward, provides a more robust and correctly-scaled training signal. By calculating a separate advantage for visual perception and textual reasoning, PDCR ensures that each skill is evaluated against its own statistical baseline, allowing visual perception steps to receive a stable, well-scaled signal that is not distorted by the predominant textual reasoning steps.

**In detail, our contributions can be summarized as follows:**

- Through a series of observations, we identify the **mixture-induced signal degradation** problem in vision-language reasoning, showing that naive, model-intrinsic rewards are **suboptimal** due to skill heterogeneity.
- We propose an **unsupervised skill decomposition** method using a **Visual Dependence Score** and optimal clustering, which requires no external labels during training.
- We introduce **PDCR**, a decomposed reward framework that uses this clustering to compute stable, intra-cluster advantages, resolving the signal degradation problem.
- We demonstrate through experiments on key vision-language reasoning benchmarks that our PDCR framework outperforms both sparse-reward baselines and the naive, global-reward formulation.

## 2. Related Works

### 2.1. Reasoning in Multimodal Models

Improving the reasoning capabilities of Multimodal Large Language Models (MLLMs) is an active area of research [4, 7, 10–12, 19, 22, 26, 27, 35, 37, 43, 45]. A common paradigm is to fine-tune MLLMs using Reinforcement Learning (RL) with verifiable rewards [9, 28, 38]. However, this approach typically relies on a sparse, outcome-based reward (i.e., whether the final answer is correct), which is inefficient for training complex, multi-step tasks.

To overcome this, several works have introduced external Process Reward Models (PRMs) [3, 21, 23, 31]. PRMs provide a dense, step-level training signal, but are notoriously costly to train, often requiring extensive and difficult dataset curation. This cost has motivated a recent shift towards finding reward signals that are model-intrinsic or self-generated.

### 2.2. Model-Intrinsic and Self-Rewarding Methods

To address the costs of external PRMs, an alternative direction is self-rewarding reinforcement learning, which sources rewards from the model’s own internal signals [44, 47, 54]. These signals often include logit-based self-certainty [53], self-confidence [14], or majority-voting [55].

A particularly relevant framework in the language domain is the Progressively Ascending Confidence Reward (PACR) [42]. PACR provides a dense, step-level reward by calculating the increase in the model’s log-probability of the ground-truth answer, guided by the inductive bias that confidence should ascend during a correct reasoning trace. This approach has proven effective for text-only reasoning but has not been sufficiently adapted for the unique challenges of multimodal tasks.

### 2.3. Decomposed Reward Modeling

A parallel line of research focuses on decomposing complex reward signals. In text-only LLMs, this is often done by evaluating a response from various perspectives, such as helpfulness, safety, or correctness [5, 33, 51]. This concept is also being explored in MLLMs, where the reward signal is often denser due to the multimodal inputs [25, 35].

However, existing work on vision-language decomposition primarily focuses on generating explicit textual descriptions (e.g., image captions) to serve as the basis for the reward [6, 15, 19]. Our work diverges from this approach by decomposing the original reasoning steps directly via an internal, process-level signal, rather than relying on the generation of additional, explicit text like an image caption.

We argue that visual and linguistic capabilities are heterogeneous skills with distinct distributions, and therefore benefit from independent normalization. Analogous concepts have been explored in concurrent works in other domains, such as multi-reward [17] or tool-use agentic tasks [56].

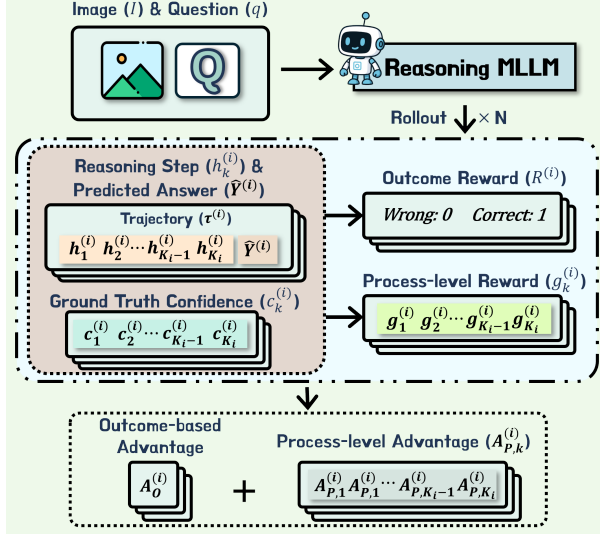


Figure 2. The baseline dense reward pipeline. For  $N$  rollouts, a sparse **Outcome Reward** ( $R^{(i)}$ ) is computed from the final answer’s correctness. Concurrently, the model’s stepwise ground-truth confidence ( $c_k^{(i)}$ ) is used to derive a dense **Process-level Reward** ( $g_k^{(i)}$ ) (i.e., the confidence gain). These two rewards are then converted into an Outcome-based Advantage ( $A_O^{(i)}$ ) and a globally-normalized Process-level Advantage ( $A_{P,k}^{(i)}$ ).

### 3. Background and Problem Setup

This section first describes the standard sparse-reward RL framework (GRPO) for reasoning training. It then discusses its sparse-reward limitations, followed by the recent, model-intrinsic signal of ground-truth confidence growth [42] which addresses them.

#### 3.1. Group Relative Policy Optimization (GRPO)

As illustrated in Figure 2, for a given visual input  $\mathbf{I}$  and question  $\mathbf{q}$ , the behavior policy  $\pi_{\theta_{\text{old}}}$  generates a group of  $N$  reasoning trajectories  $\{\tau^{(i)}\}_{i=1}^N$ . Each trajectory  $\tau^{(i)}$  consists of multiple reasoning steps  $\{h_k^{(i)}\}_{k=1}^{K_i}$ , and the final predicted answer  $\hat{Y}^{(i)}$ .

For each trajectory  $\tau^{(i)}$ , a sparse, terminal outcome reward  $R^{(i)}$  is assigned based on the final answer’s correctness:

$$R^{(i)} = \begin{cases} 1, & \text{if } \text{is\_equivalent}(\hat{Y}^{(i)}, Y_{gt}) \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

This sparse reward is then used to compute the outcome-based advantage  $A_O^{(i)}$  by normalizing it with the group’s mean and standard deviation:

$$A_O^{(i)} = \frac{R^{(i)} - \text{mean}(\{R^{(j)}\}_{j=1}^N)}{\text{std}(\{R^{(j)}\}_{j=1}^N)}. \quad (2)$$

This single, trajectory-level advantage  $A_O^{(i)}$  is then applied to all tokens within all reasoning steps  $h_k^{(i)}$  of the trajectory  $\tau^{(i)}$ . This sparse signal offers no step-specific guidance, thus exacerbating the credit assignment problem.

#### 3.2. Ground-Truth Confidence Growth as a Model-Intrinsic Process Reward

To address the limitations of this sparse, trajectory-level signal, recent work in the language domain [42] introduced a dense, model-intrinsic reward based on **ground-truth confidence growth**. This approach is guided by the inductive bias that a model’s confidence in the correct answer  $Y_{gt}$  should progressively increase as it generates a well-formed reasoning trajectory.

To operationalize this inductive bias as a computable reward, each trajectory  $\tau^{(i)}$  is first segmented into its discrete reasoning steps  $\{h_k^{(i)}\}_{k=1}^{K_i}$  (see Appendix 14 for segmentation details). As shown in the overall pipeline in Figure 2, this segmentation allows for a process-level reward. At each step  $k$  of trajectory  $\tau^{(i)}$ , the model’s confidence in the ground truth  $c_k^{(i)}$  is evaluated as its log-probability given the current prefix:

$$c_k^{(i)} = \log \pi_{\theta}(Y_{gt} | \mathbf{I}, \mathbf{q}, H_{\leq k}^{(i)}), \quad (3)$$

where  $H_{\leq k}^{(i)} = (h_1^{(i)}, \dots, h_k^{(i)})$ . The core reward signal is the *stepwise confidence gain*  $g_k^{(i)}$ , defined as the change in confidence induced by the addition of step  $h_k$ :

$$g_k^{(i)} := c_k^{(i)} - c_{k-1}^{(i)}, \quad (4)$$

where  $c_0^{(i)} = \log \pi_{\theta}(Y_{gt} | \mathbf{I}, \mathbf{q})$ . A positive gain  $g_k^{(i)} > 0$  indicates that step  $h_k$  made the ground truth more probable.

This stepwise gain  $g_k^{(i)}$  is used to compute a dense, process-level advantage. In PACR [42], the gains are first converted into a discounted return  $G_k^{(i)} = \sum_{m=k}^{K_i} \gamma^{m-k} g_m^{(i)}$ . This discounted return is then normalized against the statistics (e.g., min/max) of a global pool comprising the returns of all steps from all  $N$  trajectories. Let  $I = \{(j, k') \mid 1 \leq j \leq N, 1 \leq k' \leq K_j\}$  be the set of all step indices in the group. The process-level advantage  $A_{P,k}^{(i)}$  is:

$$A_{P,k}^{(i)} = \frac{G_k^{(i)} - \min_{(j,k') \in I} G_{k'}^{(j)}}{\max_{(j,k') \in I} G_{k'}^{(j)} - \min_{(j,k') \in I} G_{k'}^{(j)}}. \quad (5)$$

Finally, this process-level advantage is combined with the sparse, outcome-based advantage from Eq. 2:

$$A_{k,\text{total}}^{(i)} = \lambda_O A_O^{(i)} + \lambda_P A_{P,k}^{(i)}, \quad (6)$$

where  $\lambda_O$  and  $\lambda_P$  are weighting hyperparameters. During the policy update, this final step-level advantage  $A_{k,\text{total}}^{(i)}$

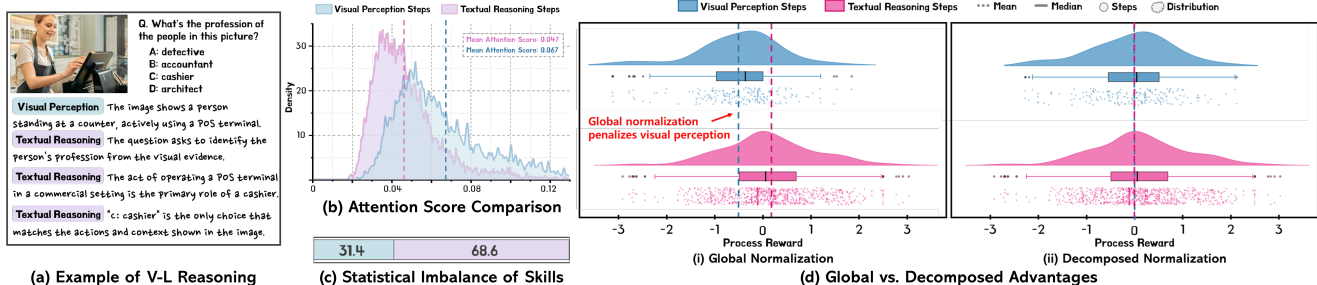


Figure 3. An illustration of our core observations and the mixture-induced signal degradation problem. (a) A V-L reasoning trace is a heterogeneous mix of **visual perception** steps and **textual reasoning** steps. (b) We validate this functional distinction by analyzing attention: **perception** steps attend highly to visual tokens, while **reasoning** steps do not. (c) These skills are statistically imbalanced: **perception** steps are sparse (31.4%), while **reasoning** steps form the dense majority (68.6%). (d) This imbalance causes *mixture-induced signal degradation*: (i) a naive, global normalization (Eq. 5) compresses and misaligns the advantage distribution for perception steps. (ii) Our proposed decomposed normalization provides a stable, well-scaled advantage for both skills.

is then applied to all tokens that comprise the reasoning step  $h_k^{(i)}$ . This combined signal provides fine-grained, step-specific guidance, effectively solving the credit assignment problem of sparse-only rewards in the language domain, while avoiding the need for external reward models.

#### 4. The Challenge of Heterogeneous Skills in Vision-Language Reasoning

While the dense reward signal from ground-truth confidence growth (Eq. 5) is effective for unimodal language tasks, its naive application to vision-language (V-L) reasoning is suboptimal. The core issue, which we validate in this section, is that V-L reasoning is not a *homogeneous* process. Instead, it is a *heterogeneous* mixture of two functionally distinct, high-level skills: (1) sparse, but critical, **visual perception** steps and (2) dense, predominant **textual reasoning** steps.

**[Observation 1]: V-L Reasoning as a Heterogeneous Skill Mix.** The two primary skills in a V-L reasoning trajectory serve different purposes. As illustrated in the example in Figure 3-(a), **visual perception** involves extracting evidence from the visual input  $I$  and translating it into textual observations. This includes identifying objects, their attributes, or their spatial relationships (e.g., “*The image shows a person standing at a counter, actively using a POS terminal.*”). In contrast, **textual reasoning** involves logical deduction, calculation, or conclusion-drawing based on previously established facts (e.g., “*The act of operating a POS terminal in a commercial setting is the primary role of a cashier.*”).

To empirically validate this functional distinction, we analyze the model’s internal attention patterns. We first annotated a representative set of reasoning trajectories, categorizing each step as either **visual perception** or **textual reasoning** (see Appendix 15 for full annotation details). We

then measured the average attention each step paid to the *input visual tokens*. As shown in the plot in Figure 3-(b), the two skills exhibit distinct attention profiles. **Visual perception** steps exhibit high attention to the input visual tokens, as they are responsible for converting visual information into a textual format. Conversely, **textual reasoning** steps attend almost exclusively to the preceding text tokens, using the information established by prior steps to perform logical inference. This shows that a V-L reasoning trajectory is not a homogeneous process, but a heterogeneous mixture of these two distinct capabilities.

**[Observation 2]: Statistical Imbalance of Skills.** Beyond their functional differences, we find a statistical imbalance between the two skills. Analyzing the same ground-truth annotated set from [Observation 1], we observe that **visual perception** steps are inherently sparse, accounting for approximately 30% of the total reasoning steps, while the **textual reasoning** steps form the dense majority (approx. 70%), as shown in Figure 3-(c). This sparsity is a key driver of the normalization problem we describe next.

**[Observation 3]: Mixture-Induced Signal Degradation from Skill Imbalance.** This statistical imbalance poses a significant challenge for the global normalization in Eq. 5. We term this issue mixture-induced signal degradation, as the global statistics (min, max) are computed from a pool overwhelmingly dominated by the returns of **textual** steps. This global baseline is, therefore, unrepresentative of the sparse **perception** steps.

Figure 3-(d) (*left*) illustrates this effect: the naive global normalization compresses and misaligns the **perception** advantage distribution, which could lead to ill-scaled credit for these critical actions. In contrast, Figure 3-(d) (*right*) shows the effect of decomposed normalization. By normalizing

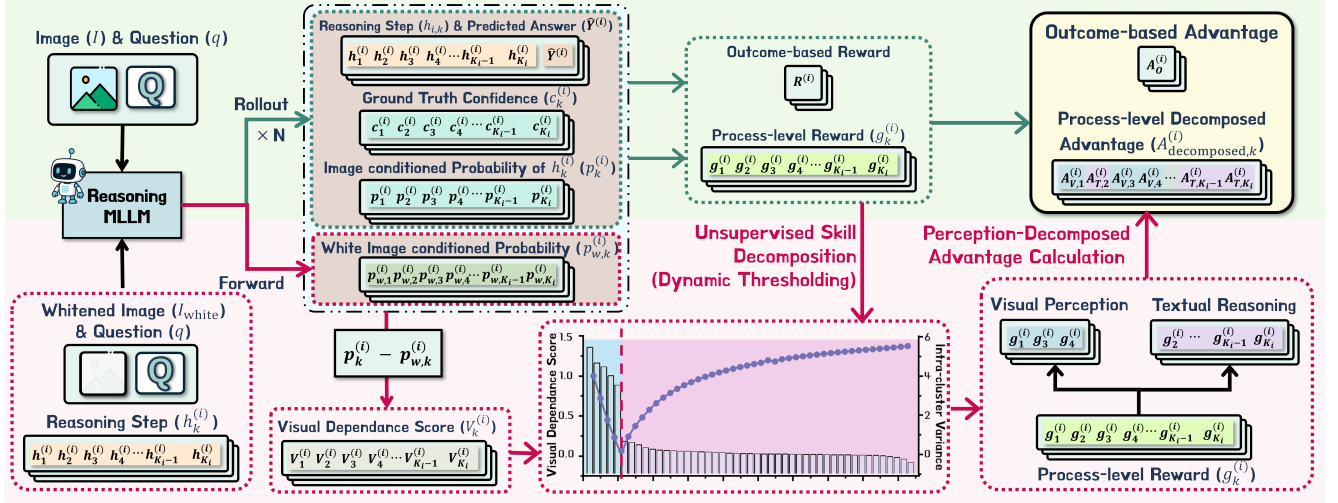


Figure 4. Overview of our PDCR (Perception-Decomposed Confidence Reward) framework. **(Top, green path)**: We compute the standard Process-level Reward ( $g_k^{(i)}$ ) (i.e., the confidence gain, Eq. 4) from the ground-truth confidence  $c_k^{(i)}$ . **(Bottom, pink path)**: We perform **Unsupervised Skill Decomposition**. We introduce a **Visual Dependence Score** ( $V_k^{(i)}$ ) (Eq. 7) by comparing the log-probability of a step given the real image ( $\mathbf{I}$ ) versus a blank image ( $\mathbf{I}_{\text{white}}$ ). We apply an optimal clustering algorithm (Otsu’s method, Eq. 9) to these scores to separate all steps  $h_k^{(i)}$  into **Visual Perception** ( $\mathcal{L}_{\text{visual}}$ ) and **Textual Reasoning** ( $\mathcal{L}_{\text{textual}}$ ) groups. **(Right)**: In the final **Perception-Decomposed Advantage Calculation**, the reward  $g_k^{(i)}$  from the top path is normalized *within* the skill cluster identified by the bottom path (Eq. 12 & 13). This yields the decomposed process advantage  $A_{\text{decomposed},k}^{(i)}$ , which is combined with the sparse outcome advantage  $A_O^{(i)}$  to produce the final training signal.

each skill cluster independently, this mixture-induced signal degradation is mitigated. The resulting advantage distributions for both skills appear better scaled and aligned, aiming to provide a more stable and representative training signal.

## 5. Method: Perception-Decomposed Confidence Reward (PDCR)

Our method, Perception-Decomposed Confidence Reward (PDCR), directly addresses the core problem identified in Section 4: that a single, homogeneous reward is suboptimal for heterogeneous V-L skills. As illustrated in Figure 4, PDCR operates in two stages. Specifically, it (i) performs **unsupervised skill decomposition** to automatically cluster steps into **visual perception** and **textual reasoning** groups, and (ii) computes a **decomposed advantage** by normalizing confidence gains *within* each respective cluster, ensuring a more stable, intra-cluster comparison.

### 5.1. Unsupervised Skill Decomposition via Visual Dependence

To decompose the reward signal during training, we must categorize each reasoning step  $h_k^{(i)}$  as **visual perception** or **textual reasoning**. While our analysis in Section 4 used externally-annotated labels (via GPT) to *validate* the problem, such labels are not available in a practical training setup. Therefore, to achieve a scalable and self-contained decom-

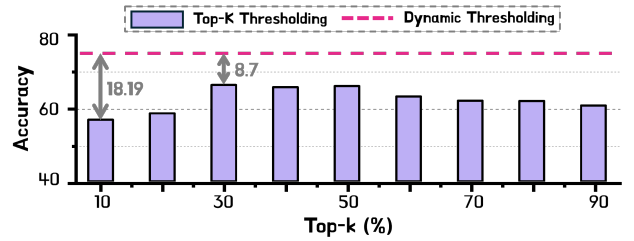


Figure 5. Our dynamic thresholding (Otsu’s method) is more accurate and robust at decomposing reasoning skills than a naive Top-K baseline. The Top-K method is highly sensitive to the  $k$  hyperparameter, peaking at 30%. Our parameter-free dynamic method significantly outperforms even the best Top-K, confirming it’s a superior approach for skill decomposition.

position, we introduce the **Visual Dependence Score**,  $V_k^{(i)}$ : a model-internal signal that quantifies a step’s reliance on the visual input.

This score quantifies how much the generation of a specific step  $h_k^{(i)}$  depended on the actual visual input  $\mathbf{I}$ , as opposed to a non-informative, blank (i.e., white) image  $\mathbf{I}_{\text{white}}$ . We define the standard log-probability of the step given the real image as  $p_k^{(i)} = \log \pi_{\theta}(h_k^{(i)} | \mathbf{I}, \mathbf{q}, H_{<k}^{(i)})$  and the white-image log-probability as  $p_{w,k}^{(i)} = \log \pi_{\theta}(h_k^{(i)} | \mathbf{I}_{\text{white}}, \mathbf{q}, H_{<k}^{(i)})$ . The Visual Dependence Score  $V_k^{(i)}$  is then the log-likelihood

ratio of these two:

$$V_k^{(i)} = p_k^{(i)} - p_{w,k}^{(i)}. \quad (7)$$

A high  $V_k^{(i)}$  score indicates that the model relied heavily on the visual input  $\mathbf{I}$  to generate the step  $h_k^{(i)}$ , marking it as a **visual perception** step. A score near zero indicates the step was primarily driven by the text prefix  $H_{<k}^{(i)}$ , marking it as a **textual reasoning** step.

To separate steps, we dynamically find an optimal threshold  $c^*$  that partitions the set of all visual dependence scores  $\mathcal{V} = \{V_k^{(i)}\}_{(i,k) \in I}$ . We determine  $c^*$  using *Otsu’s method* [24], a classic and provably optimal 1D clustering algorithm. This method is widely used in image segmentation to find a threshold that best separates a pixel intensity into two classes (e.g., foreground and background). We apply it here to our 1D distribution of visual dependence scores to find the optimal split that minimizes the total intra-cluster variance.

First, let  $\mathcal{V}_{\text{sorted}} = \{v_1, v_2, \dots, v_M\}$  be the set of all  $M = |I|$  visual dependence scores, sorted in ascending order. We then iterate through all possible split points  $k \in [1, M - 1]$ . Each  $k$  partitions the sorted data into two clusters:  $C_1(k) = \{v_1, \dots, v_k\}$  and  $C_2(k) = \{v_{k+1}, \dots, v_M\}$ .

For each potential split  $k$ , we compute the cluster means  $\mu_1(k) = \frac{1}{k} \sum_{i=1}^k v_i$  and  $\mu_2(k) = \frac{1}{M-k} \sum_{i=k+1}^M v_i$ . We then calculate the total within-cluster sum of squares (SSE) for that split:

$$SSE(k) = \sum_{i=1}^k (v_i - \mu_1(k))^2 + \sum_{i=k+1}^M (v_i - \mu_2(k))^2. \quad (8)$$

The optimal split point  $k^*$  is the one that minimizes this SSE:

$$k^* = \arg \min_k SSE(k). \quad (9)$$

The final threshold  $c^*$  is set as the value of the score at this optimal split,  $c^* = v_{k^*}$ . This threshold  $c^*$  partitions the set of all step indices  $I$  into two disjoint subsets:

$$\mathcal{I}_{\text{visual}} = \{(i, k) \in I \mid V_k^{(i)} \geq c^*\}, \quad (10)$$

$$\mathcal{I}_{\text{textual}} = \{(i, k) \in I \mid V_k^{(i)} < c^*\}. \quad (11)$$

**Validating Dynamic Thresholding.** We validate this dynamic, data-driven threshold  $c^*$  against a simpler *top-K thresholding* baseline. This naive approach simply classifies a fixed percentage  $k$  of steps with the highest  $V_k^{(i)}$  scores as **visual perception**. As shown in Figure 5, we measure the classification accuracy of both methods against the ground-truth annotated labels from Section 4. The Top-K method (purple bars) is highly sensitive to the choice of  $k$  and achieves a peak accuracy of 67.5% at  $k = 30$ . Our dynamic Otsu’s method (pink dashed line) is parameter-free

and significantly outperforms even the best Top-K baseline, achieving 76.2% accuracy. This confirms that our dynamic approach is a more robust and accurate method for separating the two skill distributions.

## 5.2. Perception-Decomposed Advantage Calculation

With the step indices now decomposed, we compute a separate, normalized advantage for each skill cluster. We take the discounted returns  $G_k^{(i)}$  (as defined in Section 3.2) and normalize them using min-max scaling, but only against other returns *from the same cluster*.

As shown in Figure 4, this yields two distinct process-level advantages. For visual steps, the advantage is:

$$A_{V,k}^{(i)} = \frac{G_k^{(i)} - \min_{(j,k') \in \mathcal{I}_{\text{visual}}} G_{k'}^{(j)}}{\max_{(j,k') \in \mathcal{I}_{\text{visual}}} G_{k'}^{(j)} - \min_{(j,k') \in \mathcal{I}_{\text{visual}}} G_{k'}^{(j)}}. \quad (12)$$

For textual steps, the advantage is:

$$A_{T,k}^{(i)} = \frac{G_k^{(i)} - \min_{(j,k') \in \mathcal{I}_{\text{textual}}} G_{k'}^{(j)}}{\max_{(j,k') \in \mathcal{I}_{\text{textual}}} G_{k'}^{(j)} - \min_{(j,k') \in \mathcal{I}_{\text{textual}}} G_{k'}^{(j)}}. \quad (13)$$

This decomposed normalization solves the “mixture-induced signal degradation” problem. The gain of a perception step is now evaluated only against its peers, resulting in a stable, well-scaled reward signal for both skills.

## 5.3. Final PDCR Advantage

Finally, the total advantage  $A_{\text{total},k}^{(i)}$  for any given step is the weighted sum of the sparse outcome advantage  $A_O^{(i)}$  (from Eq. 2) and its corresponding *decomposed* process advantage  $A_{\text{decomposed},k}^{(i)}$ :

$$A_{\text{total},k}^{(i)} = \lambda_O A_O^{(i)} + \lambda_P A_{\text{decomposed},k}^{(i)}, \quad (14)$$

where  $A_{\text{decomposed},k}^{(i)}$  is selected based on the step’s cluster:

$$A_{\text{decomposed},k}^{(i)} = \begin{cases} A_{V,k}^{(i)}, & \text{if } (i, k) \in \mathcal{I}_{\text{visual}} \\ A_{T,k}^{(i)}, & \text{if } (i, k) \in \mathcal{I}_{\text{textual}}. \end{cases} \quad (15)$$

This final advantage  $A_{\text{total},k}^{(i)}$  is then applied to all tokens comprising the step  $h_k^{(i)}$  during the policy update.

## 6. Experimental Setup

### 6.1. Models and Datasets

Our experiments utilize two open-source VLMs: **Qwen2.5-VL-3B-Instruct** and **Qwen2.5-VL-7B-Instruct**. We train

Table 1. **Main results on V-L reasoning benchmarks.** We report the accuracy across seven evaluation benchmarks. We compare our method, **PDCR (ours)**, against strong baselines, including sparse-reward (GRPO), stabilized (DAPO), and naive dense-reward (PACR) methods. The **best** score in each column is in bold, and the second-best is underlined.

Base model + Method	General Visual Understanding				Visual Math & Hallucination			Average
	MMMU-Pro	MMMU	RealWorldQA	VisNumBench	MathVerse	MathVision	HallusionBench	
Visionary-R1 (3B) by [35]	27.4	30.6	56.9	10.0	45.0	36.5	30.0	33.8
Perception-R1 (7B) by [36]	36.8	40.9	69.4	15.9	52.1	35.7	65.4	45.2
Vision-R1 (7B) by [11]	34.9	42.8	60.1	33.0	57.3	51.2	32.2	44.5
<i>Backbone model: Qwen2.5-VL-3B</i>								
Zero-shot Inference	30.5	24.0	56.7	15.4	43.7	38.6	45.5	36.3
GRPO [28]	32.8	40.4	56.6	29.9	43.5	<u>39.4</u>	<u>62.6</u>	43.6
DAPO [46]	32.6	41.0	57.5	31.8	44.0	39.1	<b>62.7</b>	44.1
PACR [42]	<b>33.4</b>	<u>41.2</u>	57.6	<u>32.9</u>	<u>45.2</u>	39.0	61.5	<u>44.4</u>
<b>PDCR (ours)</b>	<u>33.3</u>	<b>43.2</b>	<b>59.3</b>	<b>33.3</b>	<b>45.6</b>	<b>40.1</b>	61.8	<b>45.2</b>
<i>Backbone model: Qwen2.5-VL-7B</i>								
Zero-shot Inference	33.2	33.3	65.5	12.6	52.9	40.3	52.2	41.4
GRPO [28]	<u>41.9</u>	50.7	65.6	<b>41.5</b>	53.2	41.3	66.6	51.5
DAPO [46]	41.8	<u>51.0</u>	66.4	40.0	53.9	44.2	67.0	52.0
PACR [42]	41.5	50.5	<u>66.9</u>	40.1	<u>54.3</u>	<u>44.7</u>	<u>67.6</u>	<u>52.2</u>
<b>PDCR (ours)</b>	<b>42.5</b>	<b>51.5</b>	<b>67.3</b>	<u>41.1</u>	<b>55.0</b>	<b>44.8</b>	<b>68.4</b>	<b>52.9</b>

Table 2. **Ablation study on the efficacy of skill decomposition.** We compare our full PDCR method against a *Random Decomposition* baseline. This baseline randomly assigns steps to the visual or textual cluster before applying decomposed normalization. The clear performance gap validates that our data-driven *Visual Dependence Score* provides a meaningful decomposition, and that simply decomposing the reward is not sufficient.

Base model + Method	General Visual Understanding				Visual Math & Hallucination			Average
	MMMU-Pro	MMMU	RealWorldQA	VisNumBench	MathVerse	MathVision	HallusionBench	
<i>Backbone model: Qwen2.5-VL-3B</i>								
<b>PDCR (ours)</b>	<b>33.3</b>	<b>43.2</b>	<b>59.3</b>	<b>33.3</b>	<b>45.6</b>	<b>40.1</b>	61.8	<b>45.2</b>
→ Random Decomposition	33.1	41.0	57.9	31.0	44.5	39.3	<b>61.9</b>	44.1
<i>Backbone model: Qwen2.5-VL-7B</i>								
<b>PDCR (ours)</b>	<b>42.5</b>	<b>51.5</b>	<b>67.3</b>	<b>41.1</b>	<b>55.0</b>	<b>44.8</b>	<b>68.4</b>	<b>52.9</b>
→ Random Decomposition	41.6	51.2	67.1	40.6	54.6	43.7	67.2	52.3

all models on the **Vision-SR1** dataset [15]. Vision-SR1 is a diverse training corpus of approximately 47K examples, all of which include verifiable answers suitable for GRPO training. The dataset is specifically balanced to cover three key reasoning domains: Mathematical Reasoning, Common-sense Knowledge, and General Visual Understanding.

## 6.2. Evaluation Benchmarks

**General Visual Understanding.** We evaluate on **MMMU** [48], a 11.5K-question college-level benchmark for cross-disciplinary knowledge; its more rigorous successor, **MMMU-Pro** [49], which increases the difficulty; **Real-WorldQA** [34], 700 real-world images with spatially-grounded questions; and **VisNumBench** [32], a 1.9K question benchmark targeting numerical estimation and counting.

**Multimodal Mathematical Reasoning.** We use two specialized benchmarks: **MathVerse** [52], which evaluates Chain-of-Thought (CoT) correctness on 2.6K diagram-centric problems; and **MATH-Vision** [30], 3K competition-grade problems across 16 disciplines.

**Hallucination Diagnosis.** We use **HallusionBench** [8], an image-context reasoning benchmark specifically designed to test a model’s hallucination and visual illusion errors.

## 6.3. Baselines

We compare our proposed PDCR against several key baselines on both model backbones. For a direct and fair comparison, we train using identical system prompts and hyperparameters (details in Appendix 16): **GRPO** [28]; **DAPO** [46], which solves the vanishing advantage problem of GRPO (for this baseline, we add only dynamic sampling to standard GRPO, without incorporating other improvements from DAPO such as a higher clip rate.); and **PACR** [42], which applies a naive, globally-normalized ground-truth confidence growth process reward.

## 7. Experimental Results and Ablations

**Main Results.** Table 1 presents the main experimental results on both the Qwen2.5-VL-3B and Qwen2.5-VL-7B backbones. Across both model scales, our PDCR framework outperforms all baselines.

On the 7B model, PDCR achieves a final average score of 52.9, surpassing the next-best method, PACR (52.2), as well

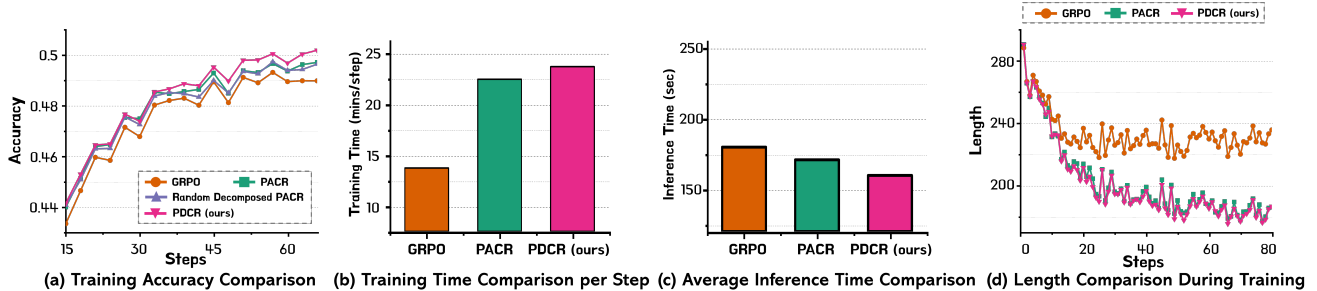


Figure 6. Training dynamics, cost, and efficiency comparison. (a) Our PDCR method (pink) converges faster and to a higher final accuracy than all baselines. (b) This improved performance comes at an expected, higher computational cost per training step. (c) & (d) This training cost is offset by a significant gain in inference-time efficiency (c). As shown in (d), this efficiency gain is driven by the fact that both PDCR and PACR learn to produce substantially shorter and more concise reasoning traces over time compared to GRPO.

as the GRPO baseline (51.5). This demonstrates the clear benefit of our decomposed reward. Notably, PDCR shows consistent improvements over the naive, globally-normalized PACR, particularly on complex understanding tasks like MMMU-Pro (42.5 vs. 41.5) and MMMU (51.5 vs. 50.5), as well as on MathVerse (55.0 vs. 54.3). This performance gain is consistent on the 3B model, where PDCR (45.2) achieves the highest average score, outperforming the naive PACR (44.4) and the sparse GRPO (43.6).

We also note a key distinction from GRPO. A well-known issue of GRPO is the **vanishing advantage problem**: if all  $N$  trajectories in a group are either correct or incorrect, the outcome advantage  $A_O^{(i)}$  becomes zero for that sample, providing no learning signal. The DAPO baseline is designed to mitigate this by re-sampling until a mixed-outcome group is obtained. Our PDCR, by definition, solves this as well, since its process-level reward  $A_{P,k}^{(i)}$  is non-zero. A key question, therefore, is whether PDCR’s gains are merely from stabilization. However, our results show that *PDCR also consistently outperforms DAPO* (e.g., 52.9 vs. 52.0 and 45.2 vs. 44.1). This comparison validates that our perception-decomposed reward provides a meaningful, high-quality training signal, and its contribution goes beyond just mitigating the vanishing advantage problem of GRPO.

**Efficacy of Skill Decomposition.** To isolate the impact of our *unsupervised skill decomposition* (Section 5.1), we compare with a **Random Decomposition** baseline.

This baseline follows the same PDCR framework, including the decomposed advantage calculation (Eq. 12 and 13). However, instead of using our Visual Dependence Score ( $V_k^{(i)}$ ) and dynamic thresholding to cluster the steps, it simply assigns each step to the  $\mathcal{I}_{\text{visual}}$  or  $\mathcal{I}_{\text{textual}}$  cluster at random.

The results are presented in Table 2. On the 7B backbone, the Random Decomposition baseline (52.3) performs significantly worse than our full PDCR method (52.9) and only slightly better than the naive PACR (52.2). This demonstrates that simply decomposing the reward is not enough.

The performance gain of our method comes from *correctly* identifying and separating the heterogeneous skills. By using a data-driven, unsupervised clustering method, our PDCR framework provides a meaningful decomposition that effectively trains both perception and reasoning.

**Training Dynamics.** We analyze the training dynamics in Figure 6-(a). The plot, which tracks average accuracy over training steps, shows that our PDCR method consistently outperforms all baselines throughout the entire training process. While the sparse GRPO baseline shows the slowest improvement, the naive PACR and our Random Decomposition baseline perform very similarly, reinforcing our earlier finding that a random decomposition offers minimal benefit.

**Training Cost and Efficiency.** We analyze computational overhead in Figure 6-(b). PACR incurs a  $\sim 1.5x$  cost increase over GRPO, as it requires an extra forward pass to compute  $c_k$  (Eq. 3). Our PDCR adds a minimal further overhead, requiring one additional forward pass per *trajectory* to compute the  $V_k$  (Eq. 7), compared to PACR. Crucially, this training cost is offset by a significant gain in generation efficiency as shown in Figure 6-(c). This gain is driven by the fact that both PACR and our PDCR learn to produce *shorter, more concise reasoning traces* over time, as shown in Figure 6-(d).

## 8. Conclusion

We introduced PDCR to address the challenge of heterogeneous skills in Vision-Language reasoning. By clustering steps via a model-internal Visual Dependence Score, PDCR decomposes the confidence growth reward in an unsupervised manner. This intra-cluster normalization resolves mixture-induced signal degradation, providing stable advantages for both perception and reasoning. Ultimately, PDCR improves strong RLVR baselines, such as GRPO and PACR, using a self-contained signal.

## 9. Acknowledgment

This research was supported by the MSIT(Ministry of Science, ICT), Korea, under the Global Research Support Program in the Digital Field program(RS-2024-00436680) supervised by the IITP(Institute for Information & Communications Technology Planning & Evaluation). This project is supported by Microsoft Research Asia.

## References

- [1] Gpt-4v(ision) system card. 2023. 1
- [2] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibao Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025. 1
- [3] Xinquan Chen, Bangwei Liu, Xuhong Wang, Yingchun Wang, and Chaochao Lu. Vrprm: Process reward modeling via visual reasoning. *arXiv preprint arXiv:2508.03556*, 2025. 1, 2
- [4] Xu Chu, Xinrong Chen, Guanyu Wang, Zhijie Tan, Kui Huang, Wenyu Lv, Tong Mo, and Weiping Li. Qwen look again: Guiding vision-language reasoning models to re-attention visual information. *arXiv preprint arXiv:2505.23558*, 2025. 2
- [5] Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Bingxiang He, Wei Zhu, Yuan Ni, Guotong Xie, Ruobing Xie, Yankai Lin, et al. Ultrafeedback: Boosting language models with scaled ai feedback. In *Forty-first International Conference on Machine Learning*. 2
- [6] Yizhuo Ding, Mingkang Chen, Zhibang Feng, Tong Xiao, Wanying Qu, Wenqi Shao, and Yanwei Fu. Vtperception-r1: Enhancing multimodal reasoning via explicit visual and textual perceptual grounding. *arXiv preprint arXiv:2509.24776*, 2025. 2
- [7] Kaixuan Fan, Kaituo Feng, Haoming Lyu, Dongzhan Zhou, and Xiangyu Yue. Sophiavl-r1: Reinforcing mllms reasoning with thinking reward. *arXiv preprint arXiv:2505.17018*, 2025. 2
- [8] Tianrui Guan, Fuxiao Liu, Xiyang Wu, Ruiqi Xian, Zongxia Li, Xiaoyu Liu, Xijun Wang, Lichang Chen, Furong Huang, Yaser Yacoob, et al. Hallusionbench: an advanced diagnostic suite for entangled language hallucination and visual illusion in large vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14375–14385, 2024. 7
- [9] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025. 1, 2, 3, 6
- [10] Siyuan Huang, Xiaoye Qu, Yafu Li, Yun Luo, Zefeng He, Daizong Liu, and Yu Cheng. Spotlight on token perception for multimodal reinforcement learning. *arXiv preprint arXiv:2510.09285*, 2025. 2, 8
- [11] Wenxuan Huang, Bohan Jia, Zijie Zhai, Shaosheng Cao, Zheyu Ye, Fei Zhao, Zhe Xu, Yao Hu, and Shaohui Lin. Vision-r1: Incentivizing reasoning capability in multimodal large language models. *arXiv preprint arXiv:2503.06749*, 2025. 1, 7
- [12] Junyeong Kim, Sunjae Yoon, Dahyun Kim, and Chang D Yoo. Structured co-reference graph attention for video-grounded dialogue. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 1789–1797, 2021. 2
- [13] Dong Bok Lee, Seanie Lee, Sangwoo Park, Minki Kang, Jinheon Baek, Dongki Kim, Dominik Wagner, Jiongdoo Jin, Heejun Lee, Tobias Bocklet, et al. Rethinking reward models for multi-domain test-time scaling. *arXiv preprint arXiv:2510.00492*, 2025. 3
- [14] Pengyi Li, Matvey Skripkin, Alexander Zubrey, Andrey Kuznetsov, and Ivan Oseledets. Confidence is all you need: Few-shot rl fine-tuning of language models. *arXiv preprint arXiv:2506.06395*, 2025. 2
- [15] 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*, 2025. 2, 7
- [16] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023. 1
- [17] Shih-Yang Liu, Xin Dong, Ximing Lu, Shizhe Diao, Peter Belcak, Mingjie Liu, Min-Hung Chen, Hongxu Yin, Yu-Chiang Frank Wang, Kwang-Ting Cheng, et al. Gdpo: Group reward-decoupled normalization policy optimization for multi-reward rl optimization. *arXiv preprint arXiv:2601.05242*, 2026. 2
- [18] Yuliang Liu, Junjie Lu, Zhaoling Chen, Chaofeng Qu, Jason Klein Liu, Chonghan Liu, Zefan Cai, Yunhui Xia, Li Zhao, Jiang Bian, et al. Adaptivestep: Automatically dividing reasoning step through model confidence. *arXiv preprint arXiv:2502.13943*, 2025. 3
- [19] Yuqi Liu, Tianyuan Qu, Zhisheng Zhong, Bohao Peng, Shu Liu, Bei Yu, and Jiaya Jia. Visionreasoner: Unified visual perception and reasoning via reinforcement learning. *arXiv preprint arXiv:2505.12081*, 2025. 2
- [20] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 6
- [21] Ruilin Luo, Zhuofan Zheng, Lei Wang, Yifan Wang, Xinzhe Ni, Zicheng Lin, Songtao Jiang, Yiyao Yu, Chufan Shi, Ruihang Chu, et al. Unlocking multimodal mathematical reasoning via process reward model. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*. 1, 2
- [22] Brandon Ong, Tej Deep Pala, Vernon Toh, William Chandra Tjhi, and Soujanya Poria. Training vision-language process reward models for test-time scaling in multimodal reasoning: Key insights and lessons learned. *arXiv preprint arXiv:2509.23250*, 2025. 2
- [23] Brandon Ong, Tej Deep Pala, Vernon Toh, William Chandra Tjhi, and Soujanya Poria. Training vision-language process reward models for test-time scaling in multimodal reasoning: Key insights and lessons learned. *arXiv preprint arXiv:2509.23250*, 2025. 1, 2
- [24] Nobuyuki Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1):62–66, 1979. 6

- [25] Övgü Özdemir and Erdem Akagündüz. Enhancing visual question answering through question-driven image captions as prompts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1562–1571, 2024. 2
- [26] Longtian Qiu, Shan Ning, Jiakuan Sun, and Xuming He. Noisygrp: Incentivizing multimodal cot reasoning via noise injection and bayesian estimation. *arXiv preprint arXiv:2510.21122*, 2025. 2
- [27] Gabriel Sarch, Snigdha Saha, Naitik Khandelwal, Ayush Jain, Michael J Tarr, Aviral Kumar, and Katerina Fragkiadaki. Grounded reinforcement learning for visual reasoning. *arXiv preprint arXiv:2505.23678*, 2025. 2
- [28] 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. 1, 2, 7, 3, 6, 9, 14
- [29] Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023. 1
- [30] Ke Wang, Junting Pan, Weikang Shi, Zimu Lu, Houxing Ren, Aojun Zhou, Mingjie Zhan, and Hongsheng Li. Measuring multimodal mathematical reasoning with math-vision dataset. *Advances in Neural Information Processing Systems*, 37:95095–95169, 2024. 7
- [31] Weiyun Wang, Zhangwei Gao, Lianjie Chen, Zhe Chen, Jinguo Zhu, Xiangyu Zhao, Yangzhou Liu, Yue Cao, Shenglong Ye, Xizhou Zhu, et al. Visualprm: An effective process reward model for multimodal reasoning. *arXiv preprint arXiv:2503.10291*, 2025. 1, 2
- [32] Tengjin Weng, Jingyi Wang, Wenhao Jiang, and Zhong Ming. Visnumbench: Evaluating number sense of multimodal large language models. *arXiv preprint arXiv:2503.14939*, 2025. 7
- [33] Zeqiu Wu, Yushi Hu, Weijia Shi, Nouha Dziri, Alane Suhr, Prithviraj Ammanabrolu, Noah A Smith, Mari Ostendorf, and Hannaneh Hajishirzi. Fine-grained human feedback gives better rewards for language model training. *Advances in Neural Information Processing Systems*, 36:59008–59033, 2023. 2
- [34] xAI. Realworldqa: Real-world spatial understanding benchmark. <https://x.ai/blog/grok-1.5v-and-realworldqa>, 2024. CC BY-ND 4.0 license. Benchmark dataset released with Grok-1.5 Vision. 7
- [35] Jiaer Xia, Yuhang Zang, Peng Gao, Yixuan Li, and Kaiyang Zhou. Visionary-r1: Mitigating shortcuts in visual reasoning with reinforcement learning. *arXiv preprint arXiv:2505.14677*, 2025. 1, 2, 7
- [36] Tong Xiao, Xin Xu, Zhenya Huang, Hongyu Gao, Quan Liu, Qi Liu, and Enhong Chen. Advancing multimodal reasoning capabilities of multimodal large language models via visual perception reward. *arXiv preprint arXiv:2506.07218*, 2025. 7
- [37] Tianrun Xu, Haoda Jing, Ye Li, Yuquan Wei, Jun Feng, Guanyu Chen, Haichuan Gao, Tianren Zhang, and Feng Chen. Defacto: Counterfactual thinking with images for enforcing evidence-grounded and faithful reasoning. *arXiv preprint arXiv:2509.20912*, 2025. 2
- [38] An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, et al. Qwen2. 5-math technical report: Toward mathematical expert model via self-improvement. *arXiv preprint arXiv:2409.12122*, 2024. 1, 2
- [39] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025. 1
- [40] Zhaohui Yang, Chenghua He, Xiaowen Shi, Linjing Li, Qiyue Yin, Shihong Deng, and Daxin Jiang. Beyond the first error: Process reward models for reflective mathematical reasoning. *arXiv preprint arXiv:2505.14391*, 2025. 3
- [41] Eunseop Yoon, Hee Suk Yoon, SooHwan Eom, Gunsoo Han, Daniel Nam, Daejin Jo, Kyoung-Woon On, Mark Hasegawa-Johnson, Sungwoong Kim, and Chang Yoo. Tlcr: Token-level continuous reward for fine-grained reinforcement learning from human feedback. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 14969–14981, 2024. 1
- [42] Eunseop Yoon, Hee Suk Yoon, Jaehyun Jang, SooHwan Eom, Qi Dai, Chong Luo, Mark A Hasegawa-Johnson, and Chang D Yoo. Pacr: Progressively ascending confidence reward for llm reasoning. *arXiv preprint arXiv:2510.22255*, 2025. 1, 2, 3, 7, 6, 9, 14
- [43] Hee Suk Yoon, Eunseop Yoon, Joshua Tian Jin Tee, Kang Zhang, Yu-Jung Heo, Du-Seong Chang, and Chang D Yoo. Bi-mdr: Bridging image history in multimodal dialogue response generation. In *European Conference on Computer Vision*, pages 378–396. Springer, 2024. 2
- [44] Hee Suk Yoon, Eunseop Yoon, Mark A Hasegawa-Johnson, Sungwoong Kim, and Chang D Yoo. Confpo: Exploiting policy model confidence for critical token selection in preference optimization. In *International Conference on Machine Learning*, pages 72641–72655. PMLR, 2025. 2
- [45] Sunjae Yoon, Dahyun Kim, Eunseop Yoon, Hee Yoon, Junyeong Kim, and Chang Yoo. Hear: Hearing enhanced audio response for video-grounded dialogue. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 11911–11924, 2023. 2
- [46] Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, YuYue, Weinan Dai, Tiantian Fan, Gaohong Liu, Juncai Liu, LingJun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Ru Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Ji-aze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Yonghui Wu, and Mingxuan Wang. DAPO: An open-source LLM reinforcement learning system at scale. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. 7, 3, 6
- [47] Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Xian Li, Sainbayar Sukhbaatar, Jing Xu, and Jason E Weston. Self-rewarding language models. In *Forty-first International Conference on Machine Learning*, 2024. 2

- [48] Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, et al. Mmmu: A massive multi-discipline multi-modal understanding and reasoning benchmark for expert agi. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9556–9567, 2024. [7](#)
- [49] Xiang Yue, Tianyu Zheng, Yuansheng Ni, Yubo Wang, Kai Zhang, Shengbang Tong, Yuxuan Sun, Botao Yu, Ge Zhang, Huan Sun, et al. Mmmu-pro: A more robust multi-discipline multimodal understanding benchmark. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15134–15186, 2025. [7](#)
- [50] Thomas Zeng, Shuibai Zhang, Shutong Wu, Christian Classen, Daewon Chae, Ethan Ewer, Minjae Lee, Heeju Kim, Wonjun Kang, Jackson Kunde, et al. Versaprm: Multi-domain process reward model via synthetic reasoning data. *arXiv preprint arXiv:2502.06737*, 2025. [3](#)
- [51] Kaiyan Zhang, Jiayuan Zhang, Haoxin Li, Xuekai Zhu, Ermo Hua, Xingtai Lv, Ning Ding, Biqing Qi, and Bowen Zhou. OpenPRM: Building open-domain process-based reward models with preference trees. In *The Thirteenth International Conference on Learning Representations*, 2025. [2](#)
- [52] Renrui Zhang, Dongzhi Jiang, Yichi Zhang, Haokun Lin, Ziyu Guo, Pengshuo Qiu, Aojun Zhou, Pan Lu, Kai-Wei Chang, Yu Qiao, et al. Mathverse: Does your multi-modal llm truly see the diagrams in visual math problems? In *European Conference on Computer Vision*, pages 169–186. Springer, 2024. [7](#)
- [53] Xuandong Zhao, Zhewei Kang, Aosong Feng, Sergey Levine, and Dawn Song. Learning to reason without external rewards. *arXiv preprint arXiv:2505.19590*, 2025. [2](#)
- [54] Yiyang Zhou, Zhiyuan Fan, Dongjie Cheng, Sihan Yang, Zhaorun Chen, Chenhang Cui, Xiyao Wang, Yun Li, Linjun Zhang, and Huaxiu Yao. Calibrated self-rewarding vision language models. *Advances in Neural Information Processing Systems*, 37:51503–51531, 2024. [2](#)
- [55] Yujun Zhou, Zhenwen Liang, Haolin Liu, Wenhao Yu, Kishan Panaganti, Linfeng Song, Dian Yu, Xiangliang Zhang, Haitao Mi, and Dong Yu. Evolving language models without labels: Majority drives selection, novelty promotes variation. *arXiv preprint arXiv:2509.15194*, 2025. [2](#)
- [56] Mingkang Zhu, Xi Chen, Bei Yu, Hengshuang Zhao, and Jiaya Jia. Stratified grpo: Handling structural heterogeneity in reinforcement learning of llm search agents. *arXiv preprint arXiv:2510.06214*, 2025. [2](#)

# PDCR: Perception-Decomposed Confidence Reward for Vision-Language Reasoning

## Supplementary Material

### Appendix Contents

10. Broader Impact	1
11. Ethics Statement	1
12. Training Procedure Pseudocode	2
13. Experimental Results on Additional Model Backbone	3
14. Segmentation Detail	3
15. Label Acquisition for Skill Analysis	4
15.1 . Annotation Setup	4
15.2 . Validation of Label Quality	4
15.3 . Qualitative Examples of Skill Decomposition	5
16. Implementation Details	6
16.1 . Training Framework and Hyperparameters	6
16.2 . Prompt Template for Training and Inference	7
17. Ablation Study on Visual Dependence Calculation for Skill Decomposition	8
18. Qualitative Comparisons of Generated Reasoning	9
19. Limitations and Future Works	14

---

### 10. Broader Impact

This work introduces a framework for improving the reasoning capabilities of multimodal Large Language Models. By leveraging the model’s intrinsic confidence dynamics, our method provides fine-grained, step-level supervision, and decomposes this signal to align with the heterogeneous skills of perception and reasoning. This is achieved without the overhead of training separate reward models or requiring manual data annotation.

### 11. Ethics Statement

This research strictly adheres to academic integrity standards, ensuring all prior work is properly cited and acknowledged. Furthermore, our experiments utilize only publicly available datasets and do not involve the collection of sensitive or personally identifiable information.

## 12. Training Procedure Pseudocode

We outline our Perception-Decomposed Confidence Reward (PDCR) training procedure in Algorithm 1. This pseudocode provides a step-by-step specification of the method summarized in Section 5. The **highlighted lines** indicate the additional processing steps introduced in our proposed PDCR compared to PACR [42].

---

### Algorithm 1: Perception-Decomposed Confidence Reward (PDCR) Training

---

```

Input : Current policy  $\pi_\theta$ , Reference policy  $\pi_{\text{ref}}$ , Training batch  $\mathcal{D} = \{(\mathbf{I}_j, \mathbf{q}_j)\}_{j=1}^B$ , White Image  $\mathbf{I}_{\text{white}}$ 
Hyperparameters : Group size  $N$ , Discount  $\gamma$ , Weights  $\lambda_O, \lambda_P$ 

// Phase 1: Data Generation & Decomposition
Initialize buffers:  $\mathcal{T} \leftarrow \emptyset, \mathcal{R} \leftarrow \emptyset$ ;
Initialize cluster pools:  $\mathcal{I}_{\text{vis}} \leftarrow \emptyset, \mathcal{I}_{\text{text}} \leftarrow \emptyset$ ;
foreach sample  $(\mathbf{I}, \mathbf{q})$  in  $\mathcal{D}$  do
  for  $i \leftarrow 1$  to  $N$  do
    Generate trajectory  $\tau^{(i)} = (h_1^{(i)}, \dots, h_{K_i}^{(i)}, \hat{Y}^{(i)}) \sim \pi_\theta(\cdot | \mathbf{I}, \mathbf{q})$ ;
    Compute outcome  $R^{(i)} \leftarrow \text{is\_equivalent}(\hat{Y}^{(i)}, Y_{gt})$ ;
    for  $k \leftarrow 1$  to  $K_i$  do
       $g_k^{(i)} \leftarrow \log \pi_\theta(Y_{gt} | \mathbf{I}, \mathbf{q}, H_{\leq k}^{(i)}) - \log \pi_\theta(Y_{gt} | \mathbf{I}, \mathbf{q}, H_{< k}^{(i)})$ ; // Conf. Gain
       $p_{\text{vis}} \leftarrow \log \pi_\theta(h_k | \mathbf{I}, \mathbf{q}, H_{< k}^{(i)})$ ;
       $p_{\text{white}} \leftarrow \log \pi_\theta(h_k | \mathbf{I}_{\text{white}}, \mathbf{q}, H_{< k}^{(i)})$ ;
       $V_k^{(i)} \leftarrow p_{\text{vis}} - p_{\text{white}}$ ; // Visual Dependence Score
    // Per-Trajectory Skill Decomposition
     $\mathbf{v}_{\text{sorted}} \leftarrow \text{Sort}(\{V_k^{(i)}\})$ 
     $c^* \leftarrow \text{OtsuMethod}(\mathbf{v}_{\text{sorted}})$ ; // Dynamic Threshold (Eq. 8)
    for  $k \leftarrow 1$  to  $K_i$  do
      if  $V_k^{(i)} \geq c^*$  then
        | Add  $(i, k)$  to  $\mathcal{I}_{\text{vis}}$ ;
      else
        | Add  $(i, k)$  to  $\mathcal{I}_{\text{text}}$ ;
    Store  $(\tau^{(i)}, \{g_k^{(i)}\})$  in  $\mathcal{T}, R^{(i)}$  in  $\mathcal{R}$ ;

// Phase 2: Decomposed Advantage Calculation
Compute sparse advantages  $A_{\text{outcome}}^{(i)}$  using  $\mathcal{R}$ ;
Compute returns  $\mathcal{G} = \{G_k^{(i)}\}$  from step gains  $\{g_k^{(i)}\}$ ;
Compute Stats:  $(\min_{\text{vis}}, \max_{\text{vis}}) \leftarrow \text{Stats}(\{G_k^{(i)} | (i, k) \in \mathcal{I}_{\text{vis}}\})$ ;
Compute Stats:  $(\min_{\text{text}}, \max_{\text{text}}) \leftarrow \text{Stats}(\{G_k^{(i)} | (i, k) \in \mathcal{I}_{\text{text}}\})$ ;
foreach step  $(i, k)$  in  $\mathcal{T}$  do
  if  $(i, k) \in \mathcal{I}_{\text{vis}}$  then
    |  $A_{\text{proc}, k}^{(i)} \leftarrow \text{Normalize}(G_k^{(i)}, \min_{\text{vis}}, \max_{\text{vis}})$ ;
  else
    |  $A_{\text{proc}, k}^{(i)} \leftarrow \text{Normalize}(G_k^{(i)}, \min_{\text{text}}, \max_{\text{text}})$ ;
   $A_{\text{total}, k}^{(i)} \leftarrow \lambda_O A_{\text{outcome}}^{(i)} + \lambda_P A_{\text{proc}, k}^{(i)}$ ;

// Phase 3: Policy Update
 $\mathcal{L}(\theta) \leftarrow -\frac{1}{B \times N} \sum \sum A_{\text{total}, k}^{(i)} \min(\dots)$ ; // GRPO Objective
 $\theta \leftarrow \theta - \alpha \nabla_\theta \mathcal{L}(\theta)$ ;

```

---

Table 3. **Experimental results on the Qwen3-VL-8B-Instruct backbone.** We report the accuracy across seven evaluation benchmarks. We compare our method, **PDCR (ours)**, against strong baselines, including sparse-reward (GRPO), stabilized (DAPO), and naive dense-reward (PACR) methods. The **best** score in each column is in bold, and the second-best is underlined.

Base model + Method	General Visual Understanding				Visual Math & Hallucination			Average
	MMMU-Pro	MMMU	RealWorldQA	VisNumBench	MathVerse	MathVision	HallusionBench	
<i>Backbone model: Qwen3-VL-8B</i>								
Zero-shot Inference	49.1	55.2	70.1	29.1	68.1	48.3	74.9	56.4
GRPO [28]	49.6	55.0	<u>71.2</u>	<u>37.2</u>	69.3	49.9	75.6	58.3
DAPO [46]	50.0	55.5	<b>71.3</b>	37.0	69.2	50.3	<u>75.8</u>	58.4
PACR [42]	<u>49.9</u>	<u>56.1</u>	70.6	37.1	<u>69.9</u>	<u>50.4</u>	75.5	<u>58.5</u>
<b>PDCR (ours)</b>	<b>50.7</b>	<b>57.1</b>	70.7	<b>37.6</b>	<b>70.6</b>	<b>51.0</b>	<b>76.0</b>	<b>59.1</b>

### 13. Experimental Results on Additional Model Backbone

We further evaluate PDCR on the recently released [Qwen3-VL-8B-Instruct](#) (implementation details are outlined in Appendix 16). As shown in Table 3, PDCR demonstrates generalization to this stronger backbone, achieving a final average score of **59.1**. This performance outperforms the sparse GRPO baseline (58.3, **+0.8%**) and also improves upon the naive dense-reward baseline, PACR (58.5, **+0.6%**).

### 14. Segmentation Detail

A prerequisite for a process-based reward framework is the segmentation of the reasoning trajectory  $\tau^{(i)}$  into a discrete sequence of steps  $\{h_k^{(i)}\}_{k=1}^{K_i}$ . The step is the fundamental unit to which a reward or advantage is assigned.

Previous work in process-reward modeling has adopted several strategies to define this unit:

- **Supervised Formatting via SFT:** Some approaches utilize Supervised Fine-Tuning (SFT) to enforce a highly structured output format, such as training the model to generate explicit step delimiters (e.g., “Step 1:”, “Step 2:”). While this simplifies segmentation, it necessitates an SFT stage to instill these formatting constraints. As detailed in our implementation (Appendix 16), our training adopts a R1-style pipeline [9] that bypasses SFT to initiate reasoning training directly via RLVR. Consequently, strategies relying on pre-learned formatting are not applicable to our setting.
- **Confidence-based Segmentation:** Some approaches leverage intrinsic model signals to dynamically segment reasoning trajectories. For instance, Liu et al. [18] proposes, a method that divides reasoning steps based on the model’s confidence in predicting the next token. This approach aims to identify true decision points by observing how the model’s uncertainty changes, avoiding arbitrary segmentation based on fixed lengths or rule-based heuristics that may not align with the model’s internal reasoning flow.
- **Rule-Based Segmentation:** A more common, lightweight, and model-agnostic approach is to use simple heuristics to parse the generated text. This is the strategy adopted by the original PACR framework [42] and other process-reward based methods [13, 40, 50]. The most common heuristic is to define a new step at each newline character ( $\backslash n$ ), or after sentence-terminating punctuation (e.g., a period followed by a space). **In this work, we adopt this strategy due to its simplicity and to ensure a fair comparison with PACR [42].**

## 15. Label Acquisition for Skill Analysis

To empirically validate the heterogeneous nature of V-L reasoning ([**Observation 1**] in Section 4) and the effectiveness of our unsupervised skill decomposition (Section 5.1), we required a set of ground truth skill labels. Since no existing dataset provides step-level distinctions between perception and reasoning, we constructed a high-quality annotated set using a state-of-the-art Large Language Model as a programmatic annotator.

### 15.1. Annotation Setup

We sampled 10,000 reasoning trajectories generated by the `Qwen2.5-VL-7B-Instruct` model from our validation set. Each trajectory was first segmented into discrete steps following Appendix 14. Subsequently, we utilized `gpt-5-2025-08-07` to classify every individual step within these trajectories. To ensure the labels captured true visual dependency rather than just semantic relevance, we designed a rigorous prompt with a specific exclusion criterion. The model was provided with the full context (Image, Question, Full Reasoning Trace) and the specific Candidate Step, and instructed to output a boolean classification in strict JSON format. The exact system instruction provided to the annotator is as follows:

#### System Instruction for Skill Annotation

**Role:** You are an expert annotator. Decide whether a SPECIFIC step in a chain-of-thought is a “visual perception” step.

**Definition:** A visual perception step should primarily involve reading or describing visual content from the provided image (e.g., recognizing objects, numbers, colors, relative positions, text in the image, bars in a chart, etc.). Steps that perform logical deductions, arithmetic, or generic reasoning without referencing observed visual features are NOT visual-perception steps.

**Exclusion Criterion:** If a step refers to visual details that could already be inferred from earlier reasoning (i.e., it does not require directly inspecting the image again), classify that step as **NOT** visual perception.

**Output Format:** Return STRICT JSON with keys exactly:

- `is_visual_perception` (boolean)
- `reason` (string)

Do not include extra keys or text outside JSON.

---

#### Input Template:

Question: `{question}`

Full thinking process (all steps concatenated): `{full.process}`

Candidate step to classify: `{step.text}`

Now output JSON only.

**Distinguishing Perception from Contextual Recall.** A critical challenge in defining these labels is separating *active perception* from *visual memory*. As noted in the **Exclusion Criterion** above, a reasoning step might reference a visual object without actually looking at the image, relying instead on facts established in previous steps. We explicitly instructed the model to classify such instances as **Textual Reasoning**, ensuring that the **Visual Perception** label is reserved strictly for steps that extract *new* information from the raw visual signal.

### 15.2. Validation of Label Quality

To verify the reliability of this automated annotation, we performed a rigorous inter-annotator agreement study:

1. **Human Inter-Annotator Agreement:** Two human experts independently annotated a random subset of 100 steps. They achieved a Cohen’s Kappa of  $\kappa = 0.82$ , indicating that the binary distinction between perception and reasoning is well-defined and unambiguous to humans.
2. **Model-Human Alignment:** We compared the primary `gpt-5` annotations against the human consensus on the same subset. The model achieved a Kappa score of  $\kappa = 0.79$  (Table 4). This high alignment confirms that the model effectively acts as a reliable proxy for human judgment, correctly adhering to the strict definitions provided in the prompt.

Table 4. **Inter-Annotator Agreement Scores.** The strong correlation ( $\kappa > 0.75$ ) validates that the labels are reliable.

Comparison Pair	Metric Interpretation	Cohen’s $\kappa$
Human Expert 1 vs. Expert 2	Task Definition Quality	0.82
GPT-5 vs. Human Consensus	Proxy Reliability	0.79

### 15.3. Qualitative Examples of Skill Decomposition

To further illustrate the functional distinction between visual perception and textual reasoning (discussed as Observation 1 in the main text), we provide a set of representative reasoning steps in Table 5. Specifically, we show examples of individual steps that we labeled as either **Visual Perception** or **Textual Reasoning** based on the annotation process described in Appendix 15.1. As shown below, perception steps are characterized by explicit grounding in the image, whereas reasoning steps rely on logical deduction or internal knowledge.

Table 5. **Qualitative Examples of Skill Decomposition.** We show examples of steps that we labeled as **Visual Perception** or **Textual Reasoning**. This separation illustrates the heterogeneous nature of the V-L reasoning task.

Sample Step Content
<p><b>EXAMPLES OF VISUAL PERCEPTION STEPS</b>  <i>(Extracting attributes, counting, reading text, identifying coordinates)</i></p>
<ul style="list-style-type: none"> <li>• The image shows a group of people in red and white attire that resembles soccer jerseys, who appear to be in distress or disappointment.</li> <li>• The use of impasto technique, where thickly applied paint creates texture, is also evident.</li> <li>• The label ‘G’ is positioned in the middle of the solar system, closer to the sun than Venus and further from the sun than Earth.</li> <li>• The individual appears to be actively paddling and navigating the river, which suggests a sense of engagement and energy.</li> <li>• Looking at the diagram, the area that is light gray and surrounds the darker gray area (umbra) is the penumbra.</li> <li>• The diagram shows a plant cell with various labeled parts.</li> <li>• This person seems to be looking at a smaller, darker figure that is separate from the main silhouette, suggesting a sense of isolation or separation.</li> <li>• The image shows a group of players standing in front of a goalpost, and there is a ball present on the field.</li> <li>• The Moon is at stage 1, and the diagram indicates that it is at the position labeled as “Midnight.” Therefore, the time of day at stage 1 is midnight.</li> </ul>
<p><b>EXAMPLES OF TEXTUAL REASONING STEPS</b>  <i>(Deduction, arithmetic, applying external knowledge, conclusion drawing)</i></p>
<ul style="list-style-type: none"> <li>• Since the question asks for the total sum of the values, we need to add the value of the blue bar (45) to the value of the red bar (30), resulting in <math>45 + 30 = 75</math>.</li> <li>• Antarctica is the only continent in the world that is entirely below the Antarctic Circle, making it stand out as the only one often highlighted in this way on world maps.</li> <li>• Substituting these into the formula for the average value, we get:</li> <li>• Combining these observations, the time shown on the clock is 5:30.</li> <li>• To find the total area of the composite shape, we must calculate the area of the square and add it to the area of the triangle.</li> <li>• Therefore, based on the architectural style and location, the most likely location of the image is Xi’an.</li> <li>• In nature, this type of relationship is known as mutualism, which is a form of symbiotic relationship.</li> <li>• Applying the Pythagorean theorem to the right-angled triangle formed by the ladder and the wall, we can solve for the unknown length of the ladder.</li> <li>• Thus, we can infer that the vehicle entering the intersection has committed a traffic violation.</li> </ul>

## 16. Implementation Details

### 16.1. Training Framework and Hyperparameters

We perform all experiments using the **EasyR1** framework. Consistent with the R1-Zero style training [9], we apply Reinforcement Learning with Verifiable Rewards (RLVR) directly on the base model, bypassing any Supervised Fine-Tuning (SFT) stage. This ensures that the reasoning behaviors we observe are genuine results of the reinforcement learning process, rather than artifacts inherited from supervised fine-tuning.

All models (including baselines) are trained on the `Vision-SR1-47K` dataset for 1 epoch. To ensure a strictly fair comparison, we maintain a consistent configuration across all methods (GRPO [28], DAPO [46], PACR [42], and our PDCR). Common hyperparameters for the optimizer, RL process, and evaluation are detailed in Table 6. All models were trained on a single node with  $8 \times$  NVIDIA A100 80GB GPUs.

**Format reward.** Given that our training pipeline bypasses SFT, the model does not inherently possess the structural priors to format its reasoning steps. To address this, we incorporate a format reward alongside the outcome accuracy reward. A trajectory is assigned a format score  $R_{\text{fmt}}$  based on the valid enclosure of the reasoning process within `<think>` and `</think>` tags, and the final answer within `\boxed{\}`. Specifically, we augment the outcome reward  $R^{(i)}$  (Eq. 1) with a fixed bonus of  $R_{\text{fmt}} = 0.1$  for compliant trajectories, while assigning  $R_{\text{fmt}} = 0$  to invalid ones.

**Details on DAPO baseline.** For the DAPO baseline [46], we adopted a specific configuration to isolate the impact of reward stability. The full DAPO framework originally introduces three modifications to GRPO: (1) *Clipping* to promote diversity and prevent entropy collapse, (2) *Dynamic Sampling* to solve the vanishing advantage problem (where a batch has 0 variance in rewards), and (3) a *Token-Level Policy Gradient* loss to mitigate length bias.

As discussed in Section 7, our primary goal in comparing with DAPO is to determine if PDCR’s performance gains are merely a result of stabilizing the training signal against the vanishing advantage problem. Therefore, our implementation of the DAPO baseline specifically applies *only* Dynamic Sampling on top of the standard GRPO framework. This controlled configuration allows for a direct, isolated comparison regarding reward signal stability.

Table 6. **Key hyperparameters** standard to the RLVR framework in EasyR1 library [?] used for training and evaluation.

Hyperparameter	Value
<i>Common Hyperparameters</i>	
Optimizer	AdamW [20]
Learning Rate	1e-6
Weight Decay	1e-2
LR Schedule	Constant
Epochs	1
Freeze Vision Encoder	False
Global Batch Size	128
Rollout Batch Size	512
Rollout Size	8
Rollout Temperature	1.0
Rollout Top-p	0.99
Max Input Prompt Length	12800
Max Response Length	2048
Max Pixels	4194304
Min Pixels	262144
Use K1 Loss	True
KL Coef	1e-2
<i>PACR &amp; PDCR Specific</i>	
$\lambda_O$	0.7
$\lambda_P$	0.3
<i>Evaluation Generation</i>	
Temperature	0.0
Max New Tokens	2048

## 16.2. Prompt Template for Training and Inference

To ensure a fair comparison, we use the same system prompt template across GRPO, DAPO, PACR, and our PDCR method as follows:

### Reasoning Template for GRPO, DAPO, PACR, and PDCR

**SYSTEM:**

You are a helpful assistant.

**USER:**

{image}

{question}

You FIRST analyze the image/video and think about the detailed, step-by-step reasoning process as an internal monologue and then provide the final answer. The reasoning process MUST BE enclosed within <think> </think> tags. The final answer MUST BE put in \boxed{ }.

## 17. Ablation Study on Visual Dependence Calculation for Skill Decomposition



Figure 7. Visual Perturbation Strategies Evaluated for Skill Decomposition. To calculate the Visual Dependence Score ( $V_k^{(i)}$ , Eq. 7), we compare the model’s probability on the (a) **Original** image against four baselines: (b) **White** (Strategy adopted in main text), (c) **Gaussian Blur**, (d) **Gaussian Noise**, and (e) **Rotate**. Our analysis confirms that strategies which effectively destroy semantic information (b, c, d) yield high decomposition accuracy, whereas simple spatial transformation (e) preserves the visual content, leading to poor separation.

In Eq. 7 of the main text, we introduced the **Visual Dependence Score** ( $V_k^{(i)}$ ) as the log-likelihood ratio between the model’s probability given the original image  $\mathbf{I}$  versus a non-informative baseline  $\mathbf{I}_{\text{white}}$ . In this section, we examine whether a blank white image the optimal baseline for measuring visual reliance.

- Following the analysis in Huang et al. [10], which examined *token-level* visual dependency, we examine the optimal strategy to divide the visual grounding steps using our dynamic thresholding. Specifically, we evaluate four distinct perturbation methods to compute the reference probability  $p_{w,k}^{(i)} = \log \pi_{\theta}(h_k^{(i)} | \mathbf{I}_{\text{perturbed}}, q, H_{<k})$ . As illustrated in Figure 7, these include:
- (b) **White Image (Strategy adopted in main text)**: Replacing the visual input with a white image to remove all information.
  - (c) **Gaussian Blur**: Applying a Gaussian blur kernel ( $\sigma = 5.0$ ) to remove high-frequency details while preserving color statistics.
  - (d) **Gaussian Noise**: Adding random Gaussian noise ( $\mu = 0, \sigma = 1$ ) to corrupt the signal structure.
  - (e) **Rotate**: Rotating the input image by  $90^\circ$  clockwise to alter spatial orientation.

**Results and Analysis.** We evaluate each perturbation strategy on the Qwen2.5-VL-7B-Instruct backbone by measuring the skill decomposition accuracy achieved using our dynamic thresholding (Eq. 9). All results are validated against the ground-truth labeled dataset described in Appendix 15. Our findings offer a clear insight into the nature of visual perception in large multimodal models:

- **Information Destruction is Necessary:** The strategies that effectively destroy visual semantics (i.e., **White**, **Gaussian Blur**, and **Gaussian Noise**) perform similarly well, achieving decomposition accuracies in the range of **75.8% - 76.2%**. This suggests that the specific *type* of noise matters less than the fact that the visual information is removed. We opted for the **White Image** in PDCR simply because it is the most computationally efficient and reproducible method to achieve this ‘blinding’ effect.
- **Spatial Transformation is Insufficient:** In contrast, the **Rotation** strategy yields significantly poorer decomposition accuracy (**65.9%**). This indicates that modern VLMs possess a degree of rotation invariance; even when the image is turned  $90^\circ$ , the model can still extract sufficient features to maintain a high likelihood for perception steps. Consequently, the log-likelihood ratio  $V_k^{(i)}$  remains low for all steps, failing to distinguish perception from reasoning.

Thus, to accurately identify if a model is ‘seeing’, we should distort the visual signal entirely rather than merely transforming it. This confirms that information-destroying baselines are essential for robust skill decomposition, validating our choice of the White Image strategy in our experiments.

## 18. Qualitative Comparisons of Generated Reasoning

In this section, we present side-by-side comparisons between our proposed PDCR and baselines (GRPO [28], PACR [42]) on V-L reasoning tasks. All examples presented were generated by the final policies trained on the Qwen2.5-VL-7B-Instruct backbone. These examples highlight the core benefit of our decomposed reward signal: the mitigation of visual hallucinations. As observed in the figures below, baseline models often exhibit a specific failure mode where they generate logically coherent reasoning chains that are premised on incorrect visual extraction (highlighted in **red**). In contrast, PDCR guides the model to accurately ground its reasoning in the visual input (highlighted in **green**). This accurate perception serves as a robust foundation for subsequent logic, leading to correct final answers in tasks involving spatial estimation, counting, and geometric reasoning.

Generated Sample 1 by GRPO



**Question:** Which of the following options is a reasonable estimate of the angle between the ground surfaces in the figure?

**Choices:** (a) 130 degrees (b) 75 degrees (c) 92 degrees (d) 42 degrees

**Ground truth:** (b) 75 degrees

**Generation:**

<think>

To estimate the angle between the ground surfaces, we need to observe the intersection of the two surfaces. The angle appears to be quite sharp, suggesting a smaller angle. Let's analyze the options:

- (a) 130 degrees: This is a very large angle, which is not typical for a ground surface intersection.
- (b) 75 degrees: This is a moderate angle, but it's still relatively large.
- (c) 92 degrees: This is a smaller angle, but it's still larger than what we typically see in ground surface intersections.
- (d) 42 degrees: This is a very small angle, which is more plausible for the intersection of two ground surfaces.

Given the visual observation, the angle appears to be closer to 42 degrees. </think>  d

Generated Sample 1 by PDCR (ours)



**Question:** Which of the following options is a reasonable estimate of the angle between the ground surfaces in the figure?

**Choices:** (a) 130 degrees (b) 75 degrees (c) 92 degrees (d) 42 degrees

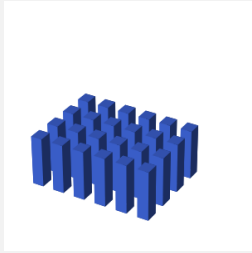
**Ground truth:** (b) 75 degrees

**Generation:**

<think>

To estimate the angle between the ground surfaces, we need to observe the intersection of the two surfaces. The angle appears to be less than 90 degrees but more than 45 degrees. The angle looks closer to 75 degrees than to 130 degrees or 42 degrees. Therefore, the most reasonable estimate is 75 degrees. </think>  b

### Generated Sample 2 by GRPO



**Question:** Which of the following options is a reasonable estimate of the number of cubes in the figure?

**Choices:** (a) 17 (b) 24 (c) 7 (d) 39

**Ground truth:** (b)

**Generation:**

<think>

To estimate the number of cubes in the figure, let's break it down step by step:

1. Visual Inspection: The figure appears to be a 3D structure with cubes stacked in a somewhat organized manner. It looks like a pyramid-like structure with a base and layers decreasing upwards.

2. Counting Layers:

- The base layer looks like it has 6 cubes.
- The next layer up has 5 cubes.
- The layer above that has 4 cubes.
- The top layer has 3 cubes.

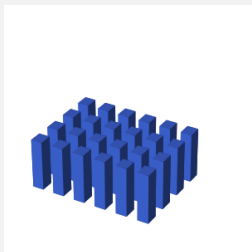
3. Summing Up: Adding these together gives us  $6 + 5 + 4 + 3 = 18$  cubes.

4. Comparison with Options:

- (a) 17 is close but slightly less.
- (b) 24 is significantly more.
- (c) 7 is much less.
- (d) 39 is much more.

Given the visual inspection and the summing up, the closest reasonable estimate is 17. </think>

### Generated Sample 2 by PDCR (ours)



**Question:** Which of the following options is a reasonable estimate of the number of cubes in the figure?

**Choices:** (a) 17 (b) 24 (c) 7 (d) 39

**Ground truth:** (b)

**Generation:**

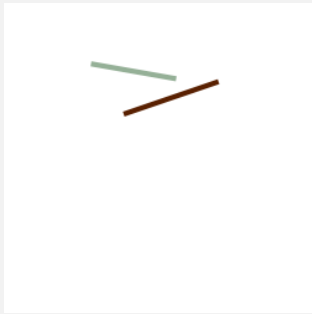
<think>

To estimate the number of cubes in the figure, we can break it down into smaller, more manageable sections. The figure appears to be a 3D structure with a grid-like pattern. By counting the cubes in one row and one column, we can estimate the total number.

- The figure has 6 rows.
- Each row appears to have 4 cubes.

So, the total number of cubes is approximately  $6 \text{ rows} * 4 \text{ cubes per row} = 24$  cubes. </think>

Generated Sample 3 by PACR



**Question:** Which of the following options is a reasonable estimate of the sharp angle between two lines in the figure?

**Choices:** (a) 67 degrees (b) 51 degrees (c) 29 degrees (d) 85 degrees

**Ground truth:** (c) 29 degrees

**Generation:**

<think>

To estimate the sharp angle between the two lines, we need to visually assess the angle formed by the intersection of the two lines.

The angle appears to be less than 90 degrees but more than 45 degrees.

- Option (a) 67 degrees is close to 90 degrees, which is not sharp. - Option (b) 51 degrees is a reasonable estimate as it is less than 90 degrees and more than 45 degrees. - Option (c) 29 degrees is too small for a sharp angle. - Option (d) 85 degrees is close to 90 degrees, which is not sharp.

Therefore, the most reasonable estimate for the sharp angle between the two lines is 51 degrees. </think>  b

Generated Sample 3 by PDCR (ours)



**Question:** Which of the following options is a reasonable estimate of the sharp angle between two lines in the figure?

**Choices:** (a) 67 degrees (b) 51 degrees (c) 29 degrees (d) 85 degrees

**Ground truth:** (c) 29 degrees

**Generation:**

<think>

To estimate the sharp angle between the two lines, we need to consider the relative orientation of the lines. The lines appear to be intersecting at a point, and the angle between them is not a right angle (90 degrees) but rather a smaller angle less than 45 degrees.

Looking at the options provided, the angle that seems to be smaller than 45 degrees and thus the most reasonable for a sharp angle between two intersecting lines is 29 degrees.

Therefore, the most reasonable estimate of the sharp angle between the two lines is 29 degrees. </think>  c

Generated Sample 4 by PACR



**Question:** Which of the following options is a reasonable estimate of the area percentage of the orange portion of the pie chart?

**Choices:** (a) (60%, 70%) (b) (80%, 90%) (c) (10%, 20%) (d) (30%, 40%)

**Ground truth:** (d) (30%, 40%)

**Generation:**

<think>

The image shows a pie chart with a single orange section. The orange section appears to cover less than a quarter of the entire circle, which is approximately 25%. Given the options, the orange section is clearly less than 30% of the pie chart. Therefore, the most reasonable estimate for the area percentage of the orange portion is (c) (10%, 20%). </think>  c

Generated Sample 4 by PDCR (ours)



**Question:** Which of the following options is a reasonable estimate of the area percentage of the orange portion of the pie chart?

**Choices:** (a) (60%, 70%) (b) (80%, 90%) (c) (10%, 20%) (d) (30%, 40%)

**Ground truth:** (d) (30%, 40%)

**Generation:**

<think>

The pie chart shows a large orange section that appears to cover more than a quarter but less than half of the circle. A quarter of a circle is 25%, and half is 50%. The orange section looks to be between 30% and 40% of the circle. Therefore, the most reasonable estimate for the area percentage of the orange portion is (30%, 40%). </think>  d

## 19. Limitations and Future Works

Our work presents a promising direction for self-rewarding V-L models, but has limitations that offer avenues for future research.

**Computational Cost.** The primary limitation of our framework is the computational overhead during the training phase. As shown in Figure 6-(b) of main paper, both PDCR and the dense-reward baseline (PACR) [42] require additional computation to estimate the step-level confidence gains (and in our case, an additional visual dependence scores). This results in a higher per-step training cost compared to standard sparse-reward methods like GRPO [28]. However, since our decomposition mechanism is lightweight, the cost of PDCR is comparable to that of PACR.

Crucially, we argued that this training overhead is a favorable trade-off for inference efficiency. As shown in Figure 6-(c)&(d), PDCR induces significantly more concise reasoning traces compared to GRPO. Consequently, while PDCR requires more GPU-hours to train, it yields a more efficient model for deployment. Future work could explore internal activation-based approximations to obtain these confidence scores without requiring full additional forward passes, potentially bridging the training cost gap.

**Binary Skill Decomposition.** Our method makes a simplifying assumption that V-L reasoning steps are a heterogeneous mixture of two *distinct* and *separable* skills. We then use Otsu’s method to enforce a single, hard threshold  $c^*$  between them. In reality, some steps may be hybrid (e.g., complex spatial reasoning that is both perceptual and logical). Our binary decomposition forces these hybrid steps into one of the two clusters, which may not be optimal. Future research could explore soft-clustering assignments or the identification of three or more distinct skill types.