# VISUALREASONINGGYM: ADAPTIVE AND VERIFIABLE DATASETS FOR COMPOSITIONAL REASONING ACROSS SCIENTIFIC AND NATURAL IMAGES

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

We experimentally identify a structural limitation of static visual datasets for training multimodal reasoning models for compositional reasoning with RLVR. Static datasets bind images and annotations to fixed questions whose complexity and learning value varies across model families and can decline during training. Yet the same images and annotations can support new questions that recover learning signal and increase the complexity of the visual reasoning task. We therefore argue that visual reasoning datasets should be more effectively represented as a generative process over the images, annotations, and reasoning operations, with questions generated at training time to match the policy, and at evaluation time at increasing difficulty until the model breaks. Building on these findings, we introduce VisualReasoningGym, a framework that repurposes existing static visual datasets and its metadata into compositional visual reasoning programs that factorizes a question into image dependent grounded atomic skills and a verifiable reasoning program. VisualReasoningGym samples multiple atomic skills per image as individual questions and composes their answers using parametric reasoning programs, ranging from aggregation and affine expressions to shortest-path and combinatorial-optimization problems. This decomposition provides exact final-answer verification and exposes intermediate outcomes. Rollout feedback adjusts compositional difficulty to the policy's evolving capabilities. We evaluate VisualReasoningGym as (1) a source of on-policy RL training data and (2) as benchmark for measuring compositional proficiency: the mean number of consecutive difficulty levels passed by a model. Continuing RLVR from Qwen3-VL-8B raises held-out pass@5 compositional proficiency from 4.55 to **6.19** of 8 levels and accuracy by **+22.3 points**, approaching the compositional proficiency of frontier models evaluated zero-shot on the same in-distribution benchmark. Training also notably improves average performance on external visual reasoning benchmarks across Qwen3-VL and Nemotron-Omni, demonstrating notable positive transfer.

## 1 INTRODUCTION

Despite recent interest in how to collect, curate, and synthesize multimodal reasoning datasets for evaluation (Wang et al., 2024; Xiao et al., 2024), SFT (Wiedmann et al., 2025; Acuna et al., 2026), RL (Chen et al., 2026; Huang et al., 2026; Sarch et al., 2026), and on-policy-distillation (Agarwal et al., 2024; Zhao et al., 2026b), the representation of visual reasoning datasets has remained structurally rigid. Inherited from the supervised era of VQA (Antol et al., 2015), a visual reasoning dataset is still commongly represented as collection of static triplets: (image, question, answer). This representation binds images and annotations to questions prescribed at dataset construction. Reinforcement learning with verifiable rewards (RLVR) challenges this. In group-relative methods such as GRPO (Shao et al., 2024), all-correct or all-incorrect rollout groups yield zero advantages. Therefore, the learning value of a question is relative to the target-evolving policy used, and "goldilock" questions producing mixed rollout for a target policy are essential for continous learning.

In this work we return to first principles, questioning how a visual reasoning dataset should be represented for RLVR. As models proliferate and exhibit different capabilities, a fixed question's utility varies drastically across model families and can decline during training. Yet, the image and



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

annotations behind it do not necessarily lose their learning value. We show that (i) the same static dataset yields mixed rollout groups for 68.4% of questions under one 7B model but only 6.2% under another of similar size; (ii) questions selected for mixed outcomes at initialization lose much of that signal within 150 training steps; and (iii) when the original question produces only correct rollouts, simply recomposing questions from the same image and annotations recovers a mixed-outcome question for 43.2% of image-annotation pairs. A fixed question can thus cease to mixed-rollout outcomes even when the image and metadata behind it still support informative reasoning problems.

Existing solutions trade one requirement for another. Questions synthesized by a model (Zhao et al., 2026a) may follow the policy but lack exact verification of the reasoning operation and, inside an RLVR loop, invite reward hacking. Program-generated questions over existing annotations (Hudson & Manning, 2019) are verifiable, but generated once and frozen into static triplets with no policy in the loop. Rendered or procedurally generated images (Johnson et al., 2017) are verifiable and could be made adaptive, but require rendering rather than reusing real images and annotations that datasets already contain. We therefore argue that visual reasoning data for RLVR should be represented as a generative process over the same images and annotations that static datasets already contain, with the question composed according to the current policy's capabilities. No image is re-rendered and no original question is discarded. Existing annotations are reused, enriched offline with extracted metadata when no available, and what changes is how the dataset and questions are represented. The image, its metadata, and a verifiable reasoning structure define a space of possible questions, and the target's policy evolving capabilities determines which question to generate.

We instantiate this abstraction in **VisualReasoningGym**, a framework that repurposes existing scientific and natural images and their annotations into environments for compositional reasoning. It samples grounded atomic visual questions and composes their answers using executable reasoning programs such that difficulty is set by the number of composed questions and the targeted reasoning operation is computed rather than annotated. Rollout feedback advances each environment's difficulty as the policy masters the current level.

## 2 The Limits of Static Visual Reasoning Data for RLVR

With binary correctness rewards, all-correct and all-incorrect response groups have zero GRPO task-reward advantages. Following the mixed-outcome criterion used by DAPO (Yu et al., 2026), we measure *Goldilocks yield*. This is the fraction of questions producing both correct and incorrect responses. Formally, for a question $q \in D$, let $c_K(q, \pi)$ count the correct responses among $K$ rollouts from policy $\pi$ we define: $\widehat{Y}_K(D, \pi) = \frac{1}{|D|} \sum_{q \in D} \mathbf{1}[0 < c_K(q, \pi) < K]$ . We use $K{=}8$ and report yield as a percentage. This is a property of a question set and a policy together, so we use it as a proxy to study (1) how much learning signal the same static questions provide across models and over training, and (2) whether recomposing questions is able to restore the image-annotation pairs.

**Static Questions Are Informative Only for Some Models.** We estimate goldilocks yield under $K{=}8$ rollouts for 500 randomly sampled items from each of three training sets: scientific figures (ArxivQA (Li et al., 2024)), natural-image VQA (A-OKVQA Schwenk et al. (2022)), and fine-grained visual search (VisualProbe-train (Lai et al., 2025)). Evaluations cover six open-weight vision-language models across three families (Qwen, Zhipu, NVIDIA). Figure 1(a) shows that A-OKVQA yields $68.4\%$ for Qwen2.5-VL-7B-Instruct but only $6.2\%$ for Qwen3-VL-8B-Instruct. Even choosing the best of the three datasets for each model leaves Qwen3-VL-8B-Instruct at $30.4\%$.

**Static Questions Can Lose Learning Value During Training.** To test whether learning value declines within a run, we evaluate Qwen3-VL-8B-Thinking checkpoints on 84 fixed questions from its training set. These questions were selected for mixed outcomes in an independent initialization calibration. Every plotted checkpoint uses fresh $K{=}8$ rollouts. Over 150 training steps, Goldilocks yield falls from $72.6\%$ to $41.7\%$, while all-correct groups rise from $22.6\%$ to $52.4\%$ (Figure 1(b)).

**Fixed Questions Leave Images and Annotations Underused.** A fixed question can lose learning signal while its image and annotations still support informative questions. To study this, we select 118 images whose original questions receive all-correct responses from Qwen3-VL-8B-Instruct in three independent groups of 8 rollouts . This is a finite-sample criterion, so a fresh group of 8 could still be mixed for some of these questions. Holding the policy, decoding strategy, images, and GT annotations fixed, question regeneration yields at least one Goldilocks question for 51 of

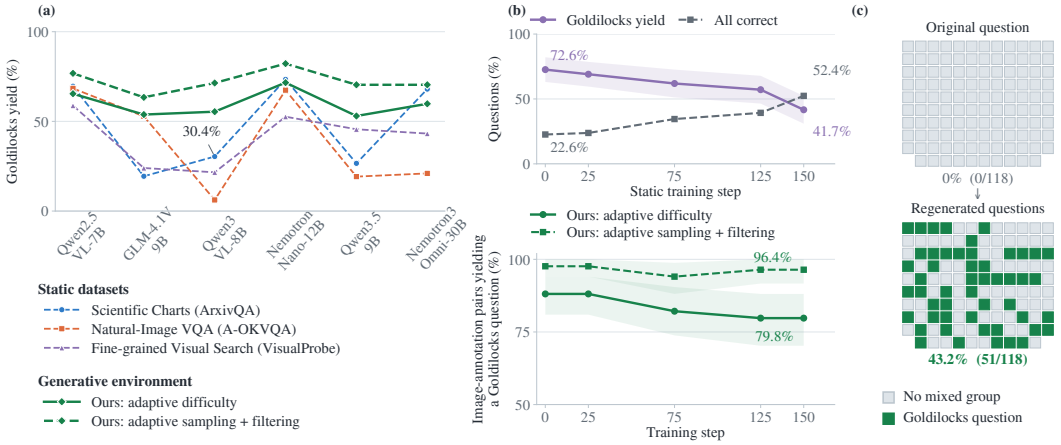


Figure 1: **The limits of static visual reasoning data for RLVR.** Goldilocks yield is the percentage of questions producing both correct and incorrect responses among $K=8$ rollouts. ( Higher is better) **(a) Static problems are informative only for some models.** We evaluate 500 fixed questions per static dataset across six models from three families, VisualReasoningGym simulations use 47 question templates. The solid green simulates a sampler that uses rollout feedback to choose difficulty for each environment, then sample problems at those levels. Adaptive sampling + filter simulates a sampler that additionally generates a larger candidate buffer and retain the groups with mixed outcomes. Both use identical per-environment allocations and a 500-problem denominator. **(b) Static problems can lose learning value during training.** (Top) We track 84 fixed training questions for Qwen3-VL-8B-Thinking. Questions were selected at the beginning for mixed outcomes in an independent calibration. Purple shows yield; gray shows all-correct groups. (Bottom) VisualReasoningGym simulations apply the two sampling strategies from (a) to the same 84 image-annotation pairs at the same checkpoints. **(c) Fixed problems leave images and annotations underused.** Keeping the policy, image, and annotations fixed, question regeneration yields at least one Goldilocks question for 51 of 118 images. Visual RLVR datasets should be represented as a process over reusable images, annotations, and reasoning operations, generating questions according to the target policy.

these images (43.2%; Figure 1(c)). The same effect appears during training: at the checkpoints of Figure 1(b), regenerating questions recovers mixed groups (Figure 1(b), bottom).

**Takeaway.**The learning value of a dataset is determined by both the questions and the policy. A static dataset binds images and their annotations to predefined questions, whose utility varies across policies and changes during training. These questions can cease to provide learning signal even when the underlying images and metadata still support informative problems. Visual reasoning data for RLVR should therefore be represented as a generative process conditioned on the current policy. The image, its metadata, and a verifiable reasoning structure define a space of possible problems. The policy's current learning frontier determines which problems to generate. In the next section, we instantiate this abstraction in VisualReasoningGym.

## 3 VisualReasoningGym: Adaptive and Verifiable Datasets for Compositional Reasoning

**Visual Datasets as a Generative Process.** A visual reasoning dataset is conventionally a set of static triplets $\mathcal{D} = (I_j, q_j, y_j)$, a representation inherited from supervised VQA. Each image is bound to the questions written at construction time. We instead represent a dataset as a collection of generators, one per image. Given an image $I$, annotations and metadata $M$, a family of verifiable reasoning operations $\mathcal{F}$, and a difficulty level $d$, a generator returns a question $q$ and its label $y$,

$$(q, y) \sim G(I, M, \mathcal{F}, d). \tag{1}$$

**Compositional Visual Reasoning.** We use the term for problems whose answer requires solving several grounded visual sub-questions on a single image and combining their answers through a reasoning operation to get to the final answer (Figure 2 bottom). The sub-questions may be independent given the image. The composition lies in the reasoning operation over their answers, which

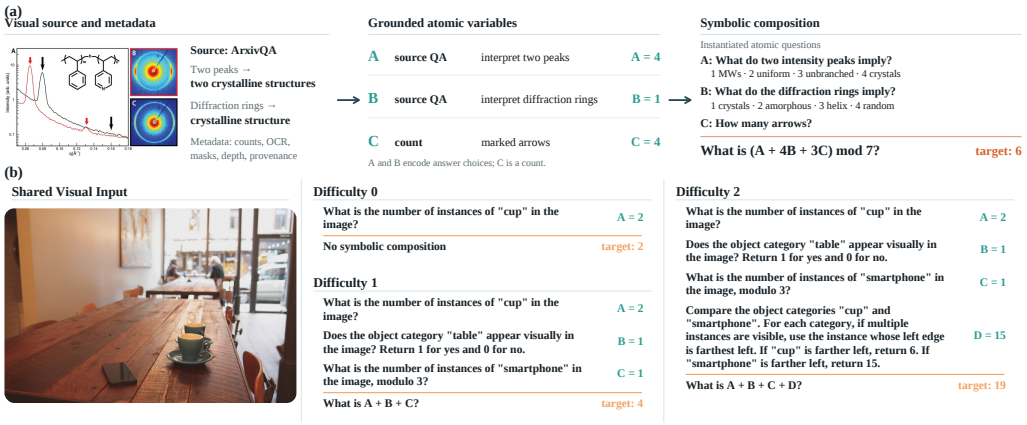


Figure 2: **VisualReasoningGym turns a static image-annotation pair into an environment. (a)** The offline stage extracts metadata $M$ from the image and its source labels once. The online stage samples atomic visual questions from $M$, each with an integer answer, and composes the answers with an executable program that computes the target. The model sees the image, the atomic questions, and the final instruction, not the intermediate answers. **(b)** On a fixed image, the difficulty $d$ sets the number of composed questions $n(d)$: D0 recovers the original question, and each higher level adds atomic questions to the same program family. During training, images and program families are resampled at every level.

the model must carry out without access to the intermediate values. Concretely, in our notation, $q$ requires solving $n(d)$ sub-questions and applying an operation from $\mathcal{F}$ to their answers. The image and annotations are those of the static dataset, enriched once offline with extracted metadata (Section 3.1). Unlike the static representation, a question is sampled rather than stored and the target is computed. Static triplets are still recovered as the special case $d{=}0$ with the original question, so no existing question is discarded. The difficulty $d$ is the interface to the policy. During training it is advanced from rollout feedback (3.2.2). During evaluation it is increased until the model fails (3.2.1). We refer to $G$ with its difficulty as an *environment* but emphasize questions are single-turn.

### 3.1 From Static Triplets to Adaptive Datasets

Given a static dataset, VisualReasoningGym builds one environment per image in two stages, offline perceptual annotation (when necessary) and online task generation with the metadata $M$ acting as the interface between perception and reasoning (Figure 2). Our instantiation of VisualReasoningGym uses 9 source datasets spanning natural scenes, counting benchmarks, scientific figures, infographics, and visual question answering (Appendix A). The framework is not tied to these sources. The offline stage requires only images, and source annotations, when present, are normalized into atomic tasks. Source questions with integer answers retain their original values. All other questions are converted. When conversion happens, we further run an independent VLM to ensure correctness across 5 trials. We keep the question if at least one trial is correct.

Formally, question generation implements $G$ in Equation 1 in two steps. First, for each of $n(d)$ positions it samples an atomic task family $k_i$ and calls its generator $g_{k_i}$, which returns a question $q_i$ with an integer answer $a_i$. Second, a symbolic program $f_\phi \in \mathcal{F}$ computes the final target (see Appendix D for examples):

$$(q_i, a_i) = g_{k_i}(I, M; \xi_i), \quad a_i \in \mathbb{Z}, \qquad y = f_\phi(a_1, \ldots, a_{n(d)}). \tag{2}$$

Here, $\xi_i$ denotes sampling choices such as the object class (e.g., `instance_count`: count that class), and $\phi$ specifies the program family and its parameters. The composed question $q$ consists of the image, the atomic questions $q_1, \ldots, q_{n(d)}$, and the final instruction.

**Perceptual Annotation Pipeline.** To populate $M$, a VLM identifies countable object classes, their instance counts, and readable words. Each predicted class is then passed to SAM 3 as an open-vocabulary prompt. We retain a class only when its predicted count matches the number of non-empty masks returned by SAM 3, discarding any mismatches. For each retained instance, we store its segmentation mask, bounding box, pixel area, and confidence score. Depth Anything 3 provides

relative depth maps for natural scenes with meaningful 3D structure. We skip depth estimation for flat media such as charts and documents. The metadata also includes the readable words and the dominant image color. Figure 2 shows how these annotations support count, depth, and referential atomic tasks. See the supplement for annotation settings.

**Atomic Visual Tasks.** Each atomic task extracts a single integer from an image. We implement 34 task families in four groups: existence and counting (presence, object counts, thresholded queries), spatial attributes (area comparison, relative depth, bounding-box overlap, position), semantic properties (text recognition, dominant and instance color), and referential reasoning (questions about objects marked by boxes, outlines, or arrows). The algorithm is not limited to those. The set is extensible. Future versions can expand on this. Task builders verify that the required annotations are available: depth tasks require depth maps, and area tasks require masks. Marker-based tasks may add a box, outline, or arrow to identify an object, but each composite problem contains at most one task that modifies the image. Questions are generated directly from the image and stored metadata without querying an online vision model.

**Symbolic Composition.** The symbolic program takes the ordered atomic integers $\mathbf{a} = [a_1, \ldots, a_n]$, generates a final natural-language prompt, and computes the ground-truth target $y = f_\phi(\mathbf{a})$. We implement 26 program families covering arithmetic, modular operations, linear comparisons, sorting, recurrences, graph queries, and constrained selection. Twenty-one families return integers, and five return real values. Each integer family can be instantiated either as a direct-answer query or as a multiple-choice question with four to six local distractors within the program's feasible output range. Real-valued families are evaluated through direct numeric answers. We emphasize this set is not restricted, and future versions can expand on this.

**Difficulty and Dataset Construction.** The difficulty parameter $d$ sets the number of atomic questions $n(d)$. At $d = 0$, the task contains one atomic question without symbolic composition. At higher levels, the model must solve and combine $d + 2$ atomic questions. Training and evaluation use disjoint sets of images. Within each split and difficulty level, we sample the 9 source datasets with roughly equal frequency. Within each source, we balance the atomic tasks and images used for creating the samples. Across difficulty levels, we fix the symbolic program family for each position in a split. The levels are therefore aligned by symbolic computation. Figure 2 reuses a single image to illustrate how the number of atomic variables increases with $d$.

### 3.2 Training and Evaluating for Compositional Reasoning

#### 3.2.1 VisualGym-Ladder: a Compositional Proficiency Benchmark

**Construction.** VisualGym-Ladder reserves 100 environments disjoint from training: 55 scientific and 45 natural images (Figure 3). Each environment fixes the image and program family and instantiates levels $d = 0, \ldots, 7$ as nested prefixes of a single reasoning chain (Figure 5). D0 is an atomic question, taken from the original source when available, while D1 through D7 compose three to nine questions. This yields 800 questions, each evaluated independently with $K$ sampled responses.

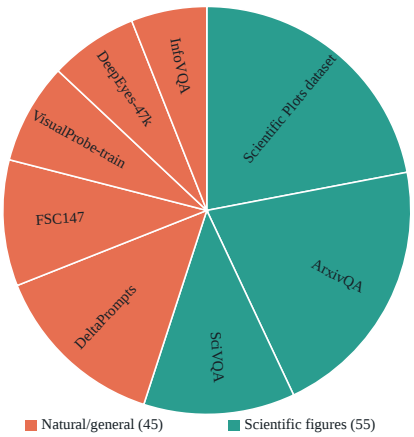


Figure 3: **Source composition of VisualGym-Ladder.** The 100 images are reused across all 8 difficulty levels.

**Metrics.** Let $c_{ids} \in \{0, 1\}$ be the correctness of response $s$ at level $d$ of environment $i$, with $N$=100 environments, $L$=8 levels, and $K$=5 responses. *Accuracy* averages correctness over all responses, $A = \frac{100}{NLK} \sum_{i,d,s} c_{ids}$. For a subset $S$ of $k$ response indices, a level is passed if any response in $S$ is correct, and $h_i(S)$ is the number of consecutive levels passed from D0. *Proficiency* at pass@$k$ is $P_k = \frac{1}{N} \sum_i \mathbb{E}_S[h_i(S)]$, ranging from 0 to $L$, and *all levels passed* is $C_k = \frac{100}{N} \sum_i \mathbb{E}_S[\mathbf{1}\{h_i(S) = L\}]$. Both expectations average over all $\binom{K}{k}$ subsets, using the same response indices at every level.

**Compositional Proficiency over Accuracy.** The first failed level locates the depth at which a model breaks, whereas accuracy averages over levels that ask for different amounts of composition. Profi-

ciency therefore measures compositional reasoning as defined in Section 3: the deepest composition on one image whose final answer a model still reaches.

**Hard, but not artificial.** The difficulty comes from the representation rather than from new images or annotations: the same environments that pose a single atomic question at D0 support compositions that the strongest available systems fail. In Table 1 we show frontier models complete the full ladder for at most 73% of environments and reach a pass@5 proficiency of 6.7 of 8 levels, while open 8B models complete 40% and reach 4.6. The spread between the two groups shows that the tasks are solvable and discriminative, and the headroom above the best frontier model shows that the ladder is not saturated.

#### 3.2.2 POLICY-ADAPTIVE TRAINING

**Hand-designed schedule.** This is the simplest setting our data enables. Static datasets do not support this. Training follows a fixed sequence of difficulty levels starting at D0, for N epochs each, carrying the model checkpoint forward between stages. Difficulty advances globally based on human-heuristics. This setting is laborious and may require tuning if the model changes.

**Adaptive difficulty.** Each environment maintains a window of up to three unlocked difficulty levels, initially containing only D0. We sample an environment uniformly, then a difficulty uniformly from its window, and draw a question at that level. To decide when to increase difficulty, we track the fraction of correct responses at each environment's highest unlocked level, accumulating results across training steps. Once accuracy reaches at least 70% over at least 16 rollouts, we unlock the next available level, reset the statistics, and retain up to three highest unlocked levels in the window. Difficulty adaptation therefore reuses the training rollouts without additional inference.

**Adaptive difficulty + oversampling and filtering.** VisualReasoningGym changes the data rather than the algorithm, so algorithmic filters such as DAPO's dynamic sampling (Yu et al., 2026) are complementary to it. The same curriculum can be extended by oversampling candidate questions at the selected difficulty levels. For each candidate, we collect a rollout group and retain the complete group only if it contains both correct and incorrect answers. A temporary buffer accumulates retained groups until the training batch is full, reusing their rollouts for the current policy update. All candidate outcomes, including discarded groups, contribute to the curriculum statistics under the same graduation rule, avoiding selection bias in difficulty advancement. This concentrates policy updates on Goldilocks questions at the cost of additional rollout generation.

## 4 EXPERIMENTS

We evaluate VisualReasoningGym as a source of RLVR training data and as a benchmark. After the setup, we report in-distribution results on VisualGym-Ladder (Table 1), transfer to 14 external benchmarks (Table 2), and an extension to another model family Nemotron-Omni (Table 3). Section 4.1 then study where the gains come from: composition, adaptation to the policy, visual grounding, and how far frontier models are from saturating the benchmark.

**Models and training.** Our primary model is Qwen3-VL-8B-Thinking. We train from its base checkpoint as well as from Qwen3-8B-DeltaThinker (Jung et al., 2026), denoted DeltaPrompt in the tables. DeltaPrompt is an 8B Qwen3-VL model trained by on-policy distillation from Qwen3-VL-235B-Thinking on synthetic prompts. We train with GRPO using answer correctness and output-format rewards, weighted 0.9 and 0.1, respectively. Both runs use 8 rollouts per prompt a temp=1.0, a lr $10^{-6}$, and a KL coeff of $10^{-3}$. For base and DeltaPrompt initializations, we use 32 and 64 prompts per batch, with response limits of 8,192 and 6,144 tokens, respectively. The curriculum unlocks the next available difficulty level when an environment reaches at least 70% cumulative accuracy over at least 16 rollouts at its highest unlocked level. Hand-design curriculum is a heuristic for control, and not a baseline. In our reference,it advances one level for epoch starting at D0.

**Baselines.** On the in-distribution held-out compositional benchmark, each baseline ablates one component of VisualReasoningGym. Text-only RLVE (Zeng et al., 2026) applies the same training recipe to text-only verifiable environments and removes the image. Direct source-task RLVR trains on the source-provided counts and questions and removes composition. D0-only trains on the same images and atomic questions at fixed difficulty and removes the curriculum. A static, hand-designed curriculum advances difficulty on a fixed step schedule and removes the dynamic adaptation to the

Table 1: **In-domain compositional reasoning on VisualGym-Ladder.** The benchmark contains 800 questions across 8 difficulty levels (D0–D7) from 100 enviroments. Accuracy is averaged over 5 sampled responses per question, with parentheses showing the sample standard deviation across 5 single-response runs. Under pass@$k$, a level is passed if any of the $k$ responses is correct. Proficiency counts the number of consecutive levels passed starting from D0, while all levels passed is the percentage of images that pass every level. Both pass@$k$ metrics average over all $k$-of-five response subsets, using the same subset across levels. Decoding setting is the same. Frontier models are for reference only and evaluated zero-shot.

| Model | Accuracy | Proficiency | | All levels passed (%) | |
|---|---|---|---|---|---|
| | (%) | pass@3 | pass@5 | pass@3 | pass@5 |
| GPT-6 Astra | 89.70 (0.30) | 6.58 | 6.59 | 72.7 | 73.0 |
| Gemini 3.8 Flash | 87.73 (0.34) | 6.56 | 6.70 | 71.0 | 73.0 |
| GPT-5.6 Sol | 86.20 (0.91) | 6.43 | 6.60 | 69.8 | 73.0 |
| Kimi K3 | 85.48 (0.58) | 6.35 | 6.48 | 67.6 | 69.0 |
| Claude Opus 5 | 82.98 (0.27) | 6.19 | 6.32 | 63.6 | 66.0 |
| Claude Opus 4.8 | 76.00 (0.52) | 5.68 | 5.91 | 55.6 | 59.0 |
| Grok 4.6 | 79.10 (0.53) | 5.66 | 5.85 | 55.3 | 58.0 |
| Qwen3-VL-8B-Instruct | 54.70 (1.93) | 3.94 | 4.57 | 30.1 | 40.0 |
| Qwen3-VL-8B-Thinking (Base) | 49.95 (1.45) | 3.95 | 4.55 | 30.6 | 40.0 |
| + RLVE (text-only RL) | 54.50 (1.39) | 4.29 | 4.78 | 32.9 | 37.0 |
| + Direct Source-Task RLVR | 55.48 (0.81) | 4.36 | 4.85 | 35.6 | 44.0 |
| + Base Questions - D0 only | 53.58 (0.96) | 4.33 | 4.83 | 33.7 | 41.0 |
| **+ Ours (hand-designed schedule)** | 65.65 (0.58) | 5.19 | 5.74 | 46.6 | 56.0 |
| **+ Ours (adaptive schedule)** | **72.25 (1.08)** | **5.79** | **6.19** | **55.5** | **64.0** |
| DeltaPrompt | 54.83 (0.40) | 4.20 | 4.93 | 32.1 | 42.0 |
| **+ Ours (adaptive schedule)** | **77.68 (1.20)** | **6.12** | **6.52** | **62.6** | **71.0** |

Figure 4: **Adaptive curriculum progression and held-out performance** for Qwen3-VL-8B-Thinking initialized from DeltaPrompt. **(a)** Environment distribution across highest unlocked difficulty levels over training steps. **(b)** Validation accuracy on fixed D7 tasks (macro-averaged over task forms with greedy decoding) and average external benchmark score across 9 of the tasks in Table 2.

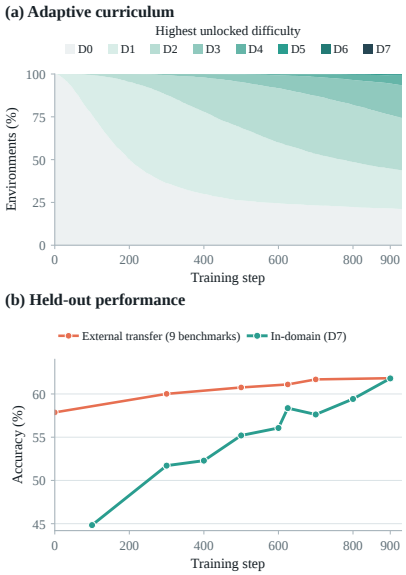


Table 2: **External transfer.** Values are mean accuracy (%). VisualProbe averages its Easy, Medium, and Hard splits. The final column reports the average across all 14 benchmark settings shown in appendix. We do not optimize for those tasks, yet performance increases in most wrt to its respective base model. Decoding settings are fixed and matched. More results and standard errors in E.

| | Fine-grained visual search | | | Scientific figures/charts | | Compositional reasoning | | |
|---|---|---|---|---|---|---|---|---|
| Method | $V^*$ Bench | VisualProbe (E/M/H avg.) | HRBench4K | CharXiv | ChartQAPro | BabyVision | ZEROBench-Sub | Average (14 benchmarks) |
| Base | 78.64 | 32.81 | 72.40 | 53.40 | 56.97 | 10.82 | 19.52 | 52.69 |
| + VisualReasoningGym | **83.46** (+4.82) | **41.43** (+8.62) | **75.83** (+3.43) | **56.64** (+3.24) | **59.20** (+2.23) | **15.64** (+4.82) | **20.72** (+1.20) | **56.34** (+3.65) |
| DeltaPrompt | 81.57 | 34.07 | 77.62 | 55.40 | 59.81 | 13.32 | **20.90** | 54.76 |
| + VisualReasoningGym | **83.35** (+1.78) | **43.34** (+9.27) | **78.18** (+0.56) | **57.70** (+2.30) | **60.52** (+0.71) | **15.79** (+2.47) | 20.66 (-0.24) | **57.47** (+2.71) |

policy (Tables 1. The DeltaPrompt checkpoint is the baseline for the distilled initialization, and proprietary frontier models, evaluated zero-shot, provide reference points.

**Evaluation protocol and metrics.** We evaluate on VisualGym-Ladder (Section 3.2.1) with 5 sampled responses per question at 8K tokens. We use exact matching and GPT-4o-mini for extraction. *Accuracy* is the % of correct responses across all questions, evaluated with 5 samples per question. *Proficiency* is computed as defined in 3.2.1. *all levels passed* is the % of trajectories that solve D0 to D7. Both pass@$k$ metrics average over all $k$-of-five response subsets. For accuracy, parentheses report the standard deviation across 5 runs. RL training settings,steps and hypers are all matched.

**Main Results.** VisualReasoningGym substantially improves compositional reasoning depth and overall accuracy (Table 1). Applied to Qwen3-VL-8B-Thinking, accuracy increases from 49.95% to 72.25%. Pass@5 proficiency rises from 4.55 to 6.19 levels, and full-ladder completion increases from 40.0% to 64.0%. Initializing from the DeltaPrompt checkpoint further raises performance. Accuracy reaches 77.68%, proficiency reaches 6.52, and full-ladder completion reaches 71.0%. Figure 5 illustrates how model D1 answers and proficiency differ between the base model and ours.

**External Transfer.** We evaluate transfer across 14 benchmark settings. Table 2 summarizes selected benchmarks and the overall average, while Table 6 in the supplement reports the complete results.

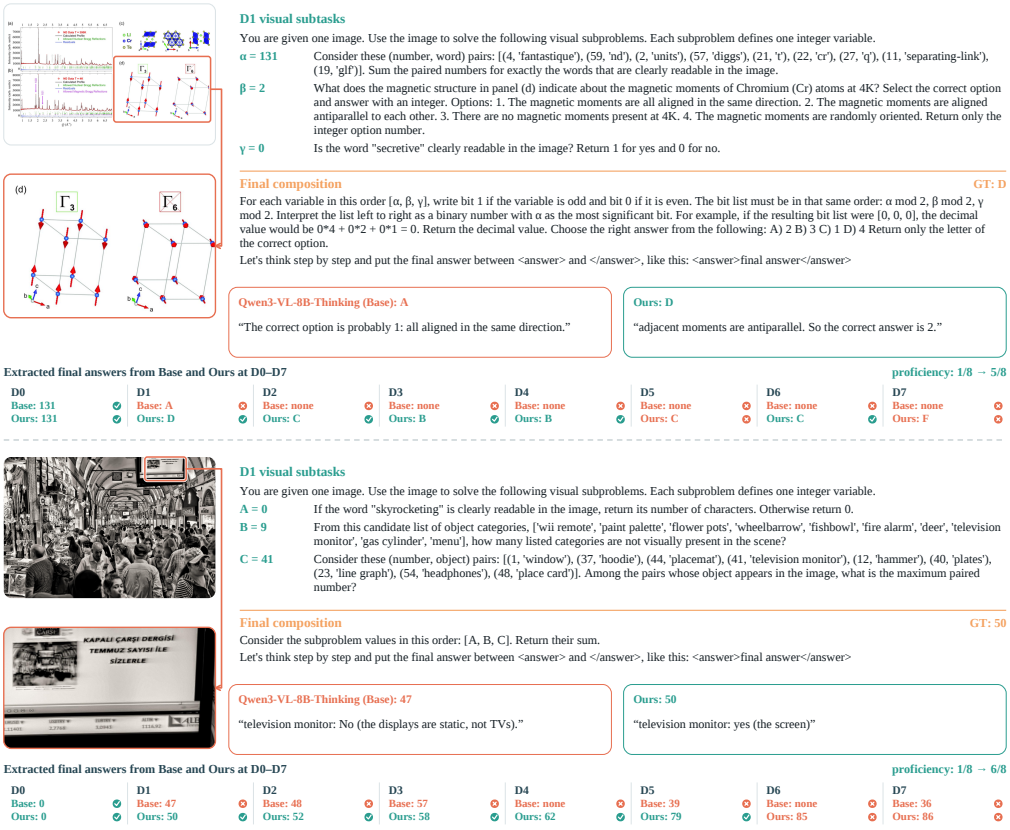


Figure 5: **Selected examples from the held-out VisualGym-Ladder.** The top panel shows a scientific figure, and the bottom panel shows a natural image. Each panel presents the D1 visual subtasks, their final composition, response excerpts from Qwen3-VL-8B-Thinking before and after dynamic VisualReasoningGym training, and correctness across D0-D7 for one sampled response. Insets highlight visual evidence. Subtask values and ground-truth final answers are shown for reference and were not given to the models. Ladder entries labeled "none" indicate responses that ended during reasoning without a final answer, upon reaching the token output limit.

We report the mean accuracy and standard error across five sampled responses per question. We observed published baseline numbers depend on inference protocols, such as the LLM judge used for answer verification, and on incompatible evaluation-framework. For fariness, we then rerun every model on every benchmark with VLMEvalKit at default settings for five independent runs. For CharXiv, where all models proved sensitive to answer extraction, GPT-6-Sol judges the full reasoning trajectory.

**Results.** As shown in Table 6 (supplement), Dynamic VisualReasoningGym achieves higher reported scores than Base on 11 of 14 settings. The largest gain is on VisualProbe Hard, which increases by 11.70 percentage points from 23.58% to 35.28%. VisualProbe Easy improves by 7.37 points, while $V^\star$Bench and BabyVision each improve by 4.82 points. RLVR VisualReasoningGym scores higher than the distilled DeltaPrompt checkpoint on 13 of the 14 comparable settings. VisualProbe Hard rises by 13.20 points from 23.40% to 36.60%, Easy rises by 8.79 points, and Medium rises by 5.82 points.

**Extension to Nemotron-Omni**. We train Nemotron-Omni for 100 steps with adaptive-difficulty. Training was done in NeMo RL, using 500 prompts per step, 16 rollouts per prompt ($T = 0.8$), and a 16,384-token combined input/output limit. This is the default continual RLVR setting provided for this model. Inference use the same token output and decoding. Pass@3 proficiency improves from 4.23 to 4.65, and pass@5 proficiency rises from 5.10 to 5.34. External transfer show gains across all five benchmarks (averaged over eight runs). (Table 3)

### 4.1 Analysis

**Does compositional training with adaptive difficulty improve on D0-only training?** Both the D0-only and full runs initialize from Qwen3-VL-8B-Thinking, use the same atomic task families,

Table 3: **Nemotron-Omni: In-Domain Performance and External Transfer.** Accuracy reports mean individual-response correctness. Proficiency is the mean number of consecutive levels passed from D0 (range: 0 to 8). All levels passed is the percentage of images for which every level is passed. External accuracies report the mean (sample SD) across eight runs evaluated using VLMEvalKit.

| Model | In-domain | | | | | External transfer (%) | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Accuracy | Proficiency | | All levels passed (%) | | $V^\star$ Bench | RWQA | CharXiv | HRBench | BabyVision |
| | (%) | pass@3 | pass@5 | pass@3 | pass@5 | | | | | |
| Nemotron-Omni | 50.53 (0.55) | 4.23 | 5.10 | 30.9 | **46.0** | 82.26 (1.17) | 75.80 (1.13) | 71.41 (0.70) | 76.02 (0.78) | 11.24 (1.42) |
| **+ RL VisualReasoningGym** | **58.20 (1.07)** | **4.65** | **5.34** | **33.2** | 41.0 | **83.38 (1.58)** | **77.03 (0.46)** | **72.66 (0.88)** | **77.67 (1.03)** | **16.43 (1.82)** |

and draw examples from a shared pool of annotated images. On the held-out ladder, the full run achieves 72.25% accuracy and 64.0% pass@5 full-ladder completion, compared to 53.58% and 41.0% for D0-only training (Table 1). **Does adapting difficulty to the policy matter?** Both schedules train on VisualReasoningGym questions. A static dataset offers no difficulty to schedule. Both improve over the base model, but the adaptive schedule is better on the held-out ladder (Table 1) and on most external settings (Table 6), and requires no per-model schedule to tune. **Do gains come from visual grounding, reasoning structure, or their combination?** Text-only RLVR trains on symbolic questions without images, reaching 54.50% accuracy. Direct Source-Task RLVR and D0-only training maintain visual input without multi-step composition, reaching 55.48% and 53.58%, respectively. The full system reaches 72.25% (Table 1). **How do frontier models perform on the VisualGym-Ladder?** Frontier models achieve between 76.00% and 89.70% accuracy on the held-out ladder (Table 1). GPT-6 Astra achieves the highest accuracy, while Gemini 3.8 Flash achieves the highest pass@5 proficiency at 6.70 of eight levels. Even with five samples per question, no frontier model completes more than 73.0% of the ladders under pass@5. Because the generator can compose additional atomic questions, future evaluations can extend the ladder beyond D7 to raise the difficulty ceiling as models advance.

## 5 Related Work

**Environment Generation and RL Training.** Reasoning Gym (Stojanovski et al., 2026) provides procedurally generated reasoning tasks with verifiable rewards and configurable difficulty for training and evaluating language models. RLVE (Zeng et al., 2026) further adapts each environment's difficulty distribution to the evolving policy. Algorithmic advances such as DAPO (Yu et al., 2026), including group-relative optimization and filtering of uninformative rollout groups, are complementary to the construction of suitable training problems. Our contribution centers on visual data. We first empirically diagnose the limitations of static questions for RLVR and develop a representation that turns existing images, annotations, and executable reasoning programs into adaptive datasets. **Visual Reasoning Data Generation** GQA (Hudson & Manning, 2019) generates questions and functional programs over real-image scene graphs, but packages these questions into a fixed dataset without policy-adaptive generation. CLEVR (Johnson et al., 2017) obtains verifiable supervision from procedurally generated scenes. Concurrent work TRON (Yang et al., 2026) incorporates such generation into adaptive visual RL environments. Both construct synthetic visual inputs. DeltaPrompts (Jung et al., 2026) synthesizes questions over existing images but relies on teacher-consistency filtering and fixes the resulting question set before training. In contrast, VisualReasoningGym reuses existing images and available annotations in a verifiable reasoning program. Policy-adaptive question generation, allows the same visual resources to support informative questions for continous learning using RLVR.

## 6 Conclusion

We show that the learning value of a visual question depends on the target policy and can diminish during training, even when its image and annotations remain useful. VisualReasoningGym represents a visual dataset as a generative process. It reuses existing datasets to compose grounded atomic questions into reasoning programs, with difficulty adapted to the policy. On VisualGym-Ladder, this raises the accuracy of the base model from 49.95% to 72.25% and its pass@5 compositional proficiency from 4.55 to 6.19 levels, with gains on most external benchmarks. The ladder can also be extend it beyond D7 providing a way to evaluate compositional reasoning as models improve.

### AI USE STATEMENT

We used generative AI tools to help with language polishing and improve clarity. We additionally use LLMs/VLMs when appropriates to generate part of the synthetic datasets as described in the method. The AI tools did not contribute to the intellectual content. We reviewed all AI-assisted edits and take responsibility for the final content of this work.

### REPRODUCIBILITY STATEMENT

We use publicly available datasets and models for our analysis and main experimental findings. Implementation details, and checkpoint selection disclosed in Section 4. Further implementation and reproducibility details are in the appendix including source dataset composition, annotation configuration, and external evaluation with standard error. Some of models use for additional metadata annotation are proprietary models served behind APIs whose updates we do not control and may differ or become unavailable if APIs change. We will release the tranining and evaluation data upon acceptance.

### ETHICS STATEMENT

Our work reuses images and annotations from publicly released datasets under their original licenses. Metadata is extracted automatically with VLMs, SAM 3, and Depth Anything 3, so even thought we did our best to ensure correctness it may contain errors that propagate to computed targets. We reduce this risk by retaining an object class only when the predicted count matches the number of segmentation masks, while recognizing that this check does not catch every error. Our results measure benchmark performance. Comparison with frontier models is zero-shot and for reference only. We do not claim we outperform production models at every visual task. The benchmark are intended to evaluate the proposed representation, not to certify models for downstream use.

## REFERENCES

David Acuna, Chao-Han Huck Yang, Yuntian Deng, Jaehun Jung, Ximing Lu, Prithviraj Ammanabrolu, Hyunwoo Kim, Yuan-Hong Liao, and Yejin Choi. Long grounded thoughts: Synthesizing visual problems and reasoning chains at scale, 2026. URL https://arxiv.org/abs/2511.05705.

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes, 2024. URL https://arxiv.org/abs/2306.13649.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pp. 2425–2433, 2015.

Ekaterina Borisova, Nikolas Rauscher, and Georg Rehm. SciVQA 2025: Overview of the first scientific visual question answering shared task. In Tirthankar Ghosal, Philipp Mayr, Amanpreet Singh, Aakanksha Naik, Georg Rehm, Dayne Freitag, Dan Li, Sonja Schimmler, and Anita De Waard (eds.), *Proceedings of the Fifth Workshop on Scholarly Document Processing (SDP 2025)*, pp. 182–210, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-265-7. URL https://aclanthology.org/2025.sdp-1.18/.

Shuang Chen, Yue Guo, Zhaochen Su, Yafu Li, Yulun Wu, Jiacheng Chen, Jiayu Chen, Weijie Wang, Xiaoye Qu, and Yu Cheng. Advancing multimodal reasoning: From optimized cold start to staged reinforcement learning, 2026. URL https://arxiv.org/abs/2506.04207.

Wenxuan Huang, Bohan Jia, Zijie Zhai, Shaosheng Cao, Zheyu Ye, Fei Zhao, Zhe Xu, Xu Tang, Yao Hu, and Shaohui Lin. Vision-r1: Incentivizing reasoning capability in multimodal large language models, 2026. URL https://arxiv.org/abs/2503.06749.

Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Justin Johnson, Bharath Hariharan, Laurens Van Der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2901–2910, 2017.

Jaehun Jung, Hyunwoo Kim, Brandon Cui, Ximing Lu, David Acuna, Prithviraj Ammanabrolu, and Yejin Choi. Deltaprompts: Escaping the zero-delta trap in multimodal distillation, 2026. URL https://arxiv.org/abs/2605.15532.

Xin Lai, Junyi Li, Wei Li, Tao Liu, Tianjian Li, and Hengshuang Zhao. Mini-o3: Scaling up reasoning patterns and interaction turns for visual search, 2025. URL https://arxiv.org/abs/2509.07969.

Lei Li, Yuqi Wang, Runxin Xu, Peiyi Wang, Xiachong Feng, Lingpeng Kong, and Qi Liu. Multimodal ArXiv: A dataset for improving scientific comprehension of large vision-language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 14369–14387, Bangkok, Thailand, aug 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.775. URL https://aclanthology.org/2024.acl-long.775.

Minesh Mathew, Viraj Bagal, Rubèn Pérez Tito, Dimosthenis Karatzas, Ernest Valveny, and C. V Jawahar. Infographicvqa, 2021. URL https://arxiv.org/abs/2104.12756.

Viresh Ranjan, Udbhav Sharma, Thu Nguyen, and Minh Hoai. Learning to count everything. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

Gabriel Sarch, Linrong Cai, Qunzhong Wang, Haoyang Wu, Danqi Chen, and Zhuang Liu. Vero: An open rl recipe for general visual reasoning, 2026. URL https://arxiv.org/abs/2604.04917.

Dustin Schwenk, Apoorv Khandelwal, Christopher Clark, Kenneth Marino, and Roozbeh Mottaghi. A-okvqa: A benchmark for visual question answering using world knowledge, 2022. URL https://arxiv.org/abs/2206.01718.

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.

Zafir Stojanovski, Oliver Stanley, Joe Sharratt, Richard Jones, Abdulhakeem Adefioye, Jean Kaddour, and Andreas Köpf. Reasoning gym: Reasoning environments for reinforcement learning with verifiable rewards. *Advances in Neural Information Processing Systems*, 38, 2026.

Zirui Wang, Mengzhou Xia, Luxi He, Howard Chen, Yitao Liu, Richard Zhu, Kaiqu Liang, Xindi Wu, Haotian Liu, Sadhika Malladi, Alexis Chevalier, Sanjeev Arora, and Danqi Chen. Charxiv: Charting gaps in realistic chart understanding in multimodal llms, 2024. URL https://arxiv.org/abs/2406.18521.

Luis Wiedmann, Orr Zohar, Amir Mahla, Xiaohan Wang, Rui Li, Thibaud Frere, Leandro von Werra, Aritra Roy Gosthipaty, and Andrés Marafioti. Finevision: Open data is all you need, 2025. URL https://arxiv.org/abs/2510.17269.

Yijia Xiao, Edward Sun, Tianyu Liu, and Wei Wang. Logicvista: Multimodal llm logical reasoning benchmark in visual contexts, 2024. URL https://arxiv.org/abs/2407.04973.

Tianze Yang, Yucheng Shi, Ruitong Sun, Jingyuan Huang, Ninghao Liu, and Jin Sun. Tron: Targeted rule-verifiable online environments for visual reasoning rl. *arXiv preprint arXiv:2606.01599*, 2026.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, et al. Dapo: An open-source llm reinforcement learning system at scale. *Advances in Neural Information Processing Systems*, 38:113222–113244, 2026.

Zhiyuan Zeng, Hamish Ivison, Yiping Wang, Lifan Yuan, Shuyue Stella Li, Zhuorui Ye, Siting Li, Jacqueline He, Runlong Zhou, Tong Chen, Chenyang Zhao, Yulia Tsvetkov, Simon Shaolei Du, Natasha Jaques, Hao Peng, Pang Wei Koh, and Hannaneh Hajishirzi. Rlve: Scaling up reinforcement learning for language models with adaptive verifiable environments. In *International Conference on Machine Learning (ICML)*, 2026.

Andrew Zhao, Yiran Wu, Tong Wu, Quentin Xu, Yang Yue, Matthieu Lin, Shenzhi Wang, Qingyun Wu, Zilong Zheng, and Gao Huang. Absolute zero: Reinforced self-play reasoning with zero data. *Advances in Neural Information Processing Systems*, 38:105816–105879, 2026a.

Siyan Zhao, Zhihui Xie, Mengchen Liu, Jing Huang, Guan Pang, Feiyu Chen, and Aditya Grover. Self-distilled reasoner: On-policy self-distillation for large language models, 2026b. URL https://arxiv.org/abs/2601.18734.

Ziwei Zheng, Michael Yang, Jack Hong, Chenxiao Zhao, Guohai Xu, Le Yang, Chao Shen, and Xing Yu. Deepeyes: Incentivizing "thinking with images" via reinforcement learning, 2026. URL https://arxiv.org/abs/2505.14362.

Table 4: **Visual content across source datasets.** Sources cover natural scenes, scientific figures, mixed visual reasoning, counting scenes, infographics, and documents.

| Source dataset | Visual content |
|---|---|
| A-OKVQA | Natural scenes |
| ArxivQA | Scientific figures |
| DeepEyes-Datasets-47k | Mixed visual reasoning |
| DeltaPrompts | Mixed visual reasoning |
| FSC147 | Natural counting scenes |
| InfoVQA | Infographics and documents |
| SciVQA | Scientific figures |
| VisualProbe | Mixed visual reasoning |
| Scientific Plots | Private collection of matplotlib figures |

Table 5: **Evaluation task composition.** Distribution of visual subtasks and final compositional operations across the 100 difficulty-7 evaluation trajectories. Each trajectory contains nine visual subtasks and one final operation.

| Category | Count | Share |
|---|---|---|
| *Visual subtasks* | | |
| Text and OCR operations | 550 | 61.1% |
| Source-provided visual QA | 141 | 15.7% |
| Object counting, presence, area, and spatial relations | 107 | 11.9% |
| Color recognition and comparison | 102 | 11.3% |
| **Total** | **900** | **100.0%** |
| *Final compositional operations* | | |
| Aggregation and ordering | 46 | 46.0% |
| Arithmetic, algebra, and logic | 30 | 30.0% |
| Optimization and combinatorics | 24 | 24.0% |
| **Total** | **100** | **100.0%** |

# A Source Dataset Composition

We draw training images from nine sources: A-OKVQA Schwenk et al. (2022), ArxivQA Li et al. (2024), DeepEyes-Datasets-47k Zheng et al. (2026), DeltaPrompts Jung et al. (2026), FSC147 Ranjan et al. (2021), InfoVQA Mathew et al. (2021), SciVQA Borisova et al. (2025), VisualProbe Lai et al. (2025) (only the training split), and Scientific Plots. At each difficulty level, the planner samples these sources with roughly equal frequency.

The train and evaluation pools are disjoint at the image level. Reusing an image within one split is allowed because an annotation can support many task compositions. Least-used eligible image sampling spreads this reuse across the available pool.

# B Implementation and Reproducibility Details

This section records the settings that determine annotations, task eligibility, and dataset composition.

## B.1 Annotation Configuration

We use GPT-5.5 for perceptual annotation and for converting source questions to integer-valued supervision. Images are resized to at most 1,600 pixels on their longest side before inference. The maximum response length is 1,200 tokens. The annotation prompt requests at most 32 discrete object categories and at most 64 visible instances per category.

SAM 3 performs open-vocabulary instance segmentation with confidence threshold 0.5. We retain a predicted category only when its VLM count equals its number of nonempty SAM 3 masks. Accepted masks determine instance bounding boxes and visible areas. Source-provided counts are merged afterward and remain valid without masks.

For sources with meaningful scene depth, we run `depth-anything/DA3-SMALL` at processing resolution 504. We use relative depth only for comparisons within object masks. We do not treat it as metric distance. Depth is omitted when a dataset sidecar disables it and for sources whose images do not support meaningful scene depth.

We estimate the dominant image color after applying orientation metadata and converting the image to RGB. The image is thumbnailed to fit within $160\times160$, quantized to eight colors with median-cut quantization, and assigned its most frequent color.

### B.2 Converted-Question Screening

We assess converted questions with `qwen3-vl-235b-a22b-instruct`. Each question receives five trials at temperature 0.7. The assessor sees an image resized to at most 1,280 pixels on its longest side. We retain a question if at least one trial is correct. DeltaPrompts uses source-provided teacher-consensus answers and does not require this additional assessment.

For multiple-choice source questions, the sidecar stores the correct option and distractors separately. At task generation time, a seeded random generator reshuffles the options and recomputes the one-based target index. The answer position therefore changes across samples to avoid bias.

### B.3 Task Eligibility and Image Rendering

Each atomic family declares the evidence it requires. Count-only supervision can instantiate counting tasks. Area, position, region, instance-color, and depth tasks require the corresponding geometric metadata. Negative object and word candidates (distractors) come from vocabularies collected across the annotated corpus.

Six active families may modify the rendered image: `boxed_object_index`, `outlined_object_index`, `pointed_object_index`, `marked_instance_region_index`, `object_dominant_color_index`, and `object_dominant_color_pair`. A problem contains at most one such family. In the bottom row of Figure 2, for example, the `outlined_object_index` atomic task outlines an oyster to identify the queried object.

### B.4 Dataset Construction

We use difficulty levels $d = 0, \ldots, 7$. Each level is planned independently with a fixed seed. Rows at the same index retain the same symbolic environment family across levels. Their images and atomic tasks are sampled independently. This keeps the reasoning-family schedule stable while increasing visual and task diversity.

Before row generation, images are assigned to source-stratified, disjoint training and evaluation pools. The split and all task choices use seed 0. A cached support index records every atomic family available for each image. For each difficulty, the planner selects source datasets with near-uniform frequency and favors atomic families with the lowest current count. It then chooses among the least-used images that support the selected task set.

Datasets with normalized source questions contribute at least one `ground_truth_numeric_question` to every selected row. When an image contains several valid questions, the planner fills as many atomic slots as possible with distinct source questions before adding generated tasks. The planner allows at most one image-modifying family in the remaining slots.

The symbolic environment order cycles through all 47 registered forms: 26 direct-answer families and 21 automatically generated multiple-choice forms. This gives each integer-valued base family near-equal direct and multiple-choice representation. The five real-valued (float answers) families remain direct answer only.

## C Visual Atomic Tasks

Each atomic task asks a question about a single image and returns an integer answer derived from source supervision or image metadata. For example, `instance_count` asks for the annotated count of an object category:

```
What is the number of instances of "egg" in the image?
```

The `ground_truth_numeric_question` task reuses questions from the source datasets. The examples below show one sample prompt from each source dataset.

**A-OKVQA**

```
What time does the analog clock read?  Express the time as a four-digit
integer in HHMM format, with no colon, and return only the integer.
```

**ArxivQA**

```
Which layer in figure (b) has the largest thickness?  Select the
correct option and answer with an integer.
Options:
1.  Ta-C60
2.  C60
3.  Fe-SiO
4.  Fe
Return only the integer option number.
```

**DeepEyes-Datasets-47k**

```
Find the measure of angle JKL in the image.  Return the measure as an
integer number of degrees.
```

**DeltaPrompts**

```
What is the sum of the numerical digits visible on the car's license
plate?  Return the final answer as an integer.
```

**FSC147**

```
Count the number of clams in the image.  Return the final answer as an
integer.
```

**InfoVQA**

```
What is the IDI rank of the UK in 2012?  Return the rank as an integer.
```

**SciVQA**

```
What percentage of the pie chart is orange?  Return the percentage as
an integer without the percent sign.
```

**VisualProbe (train split)**

```
Is there a tree to the right of the house with the red roof?  Answer 1
for yes and 0 for no.  Return only the integer.
```

Other atomic tasks evaluate different visual cues. `visible_word_count` counts readable words from a candidate list. `marked_instance_region_index` marks an object instance with an arrow and returns the one-based index of the listed image quadrant containing most of its segmented pixels. `object_depth_extreme_choice` compares the nearest or farthest masked depth points of two object categories and returns the sampled integer assigned to the category satisfying the requested relation.

## D Final Symbolic Composition Tasks

Final composition tasks compute the target value by executing a program over the integer answers to visual atomic questions. For example, `affine_expression` combines multiple atomic answers, e.g., A, B, and C, using sampled coefficients and a constant:

```
Final task:  Compute the value of this arithmetic expression:  -2*A -
3*B - 2*C + 2.
```

Other composition tasks use atomic answers differently. `weighted_graph_shortest_path` sets edge costs in a small undirected graph from atomic answers and returns the shortest-path cost.

Table 6: **External Transfer and Generalization to Existing Benchmarks.** Scores report mean accuracy (standard error) across five samples per question. Background colors mark the best and second-best. RWQA denotes RealWorldQA, and CharXiv uses the reasoning validation split.

| Method | $V^\star$ Bench | RWQA | MMVP | LogicVista | VisualProbe-E | VisualProbe-M | VisualProbe-H |
|---|---|---|---|---|---|---|---|
| Base | 78.64 (0.61) | 72.92 (0.21) | 59.20 (1.06) | 64.07 (0.79) | 48.94 (1.21) | 25.90 (0.59) | 23.58 (0.73) |
| +Hand-designed curriculum | 80.84 (0.61) | 74.90 (0.43) | 60.13 (0.93) | 65.46 (0.66) | 49.50 (1.57) | 27.91 (0.36) | 26.23 (1.62) |
| + Dynamic VisualReasoningGym | **83.46 (0.21)** | **75.82 (0.22)** | **64.00 (0.37)** | 63.22 (0.49) | 56.31 (0.73) | **32.69 (0.76)** | 35.28 (1.14) |
| DeltaPrompt | 81.57 (0.42) | 75.16 (0.13) | 62.00 (1.38) | 67.11 (0.77) | 52.91 (0.17) | 25.90 (0.54) | 23.40 (1.17) |
| +RLVR VisualReasoningGym | 83.35 (0.42) | 75.42 (0.31) | 62.13 (0.71) | **67.34 (0.37)** | **61.70 (0.78)** | 31.72 (0.54) | **36.60 (0.63)** |

| Method | CharXiv | ZEROBench-sub | MMStar | MathVista-mini | ChartQAPro | HRBench4K | BabyVision |
|---|---|---|---|---|---|---|---|
| Base | 53.40 (1.29) | 19.52 (0.51) | 73.04 (0.22) | 78.26 (0.32) | 56.97 (0.22) | 72.40 (0.40) | 10.82 (0.59) |
| +Hand-designed curriculum | 53.80 (1.32) | 20.72 (0.94) | **74.25 (0.30)** | 78.98 (0.33) | 56.55 (0.13) | 74.60 (0.30) | 13.35 (0.82) |
| + Dynamic VisualReasoningGym | 56.64 (1.33) | 20.72 (0.70) | 72.20 (0.15) | 77.74 (0.27) | 59.20 (0.18) | 75.83 (0.32) | 15.64 (0.29) |
| DeltaPrompt | 55.40 (1.31) | **20.90 (0.42)** | 72.67 (0.16) | 78.88 (0.40) | 59.81 (0.21) | 77.62 (0.35) | 13.32 (0.14) |
| +RLVR VisualReasoningGym | **57.70 (0.36)** | 20.66 (0.59) | 73.95 (0.22) | **79.52 (0.42)** | **60.52 (0.32)** | **78.18 (0.24)** | **15.79 (0.36)** |

`knapsack_value` uses atomic answers to define item weights, values, and capacity, returning the maximum feasible value. `fractional_selection_ratio` selects a fixed number of items to maximize the ratio of total numerators to total denominators and returns the rounded result.

## E Additional External Results

Table 2 summarizes selected external benchmark results. Table 6 extends the comparison to all 14 benchmark settings and includes the hand-designed curriculum baseline.